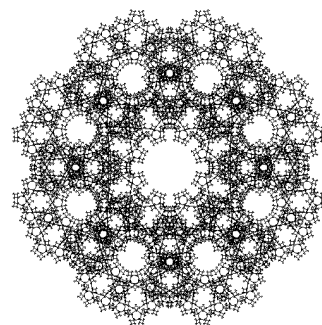
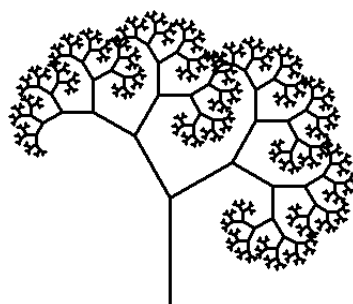
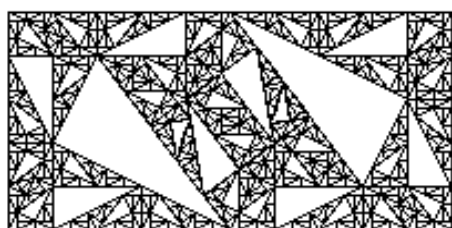


# Logo versenyfeladatok

ABONYI-TÓTH ANDOR, HEIZLERNÉ BAKONYI VIKTÓRIA, ZSAKÓ LÁSZLÓ

ELTE INFORMATIKAI KAR, NJSZT TEHETSÉGGONDOZÁSI SZAKOSZTÁLY



**SZÉCHENYI** 2020

2020

**Európai Unió**  
Európai Szociális  
Alap



**BEFEKTETÉS A JÖVŐBE**

## Bevezető

A Logo Országos Számítástechnikai Tanulmányi Verseny elindítását az országban az Informatika-Számítástechnika Tanárok Egyesülete által kezdeményezett és lebonyolított Comenius Logo akció tette lehetővé. Emiatt a Logo tanítása rohamosan terjedt, s felmerült az igény, hogy a Nemes Tihamér OKSzTV-től függetlenül, önálló Logo verseny induljon.

Az 1997/98-as tanévben kísérletképpen indult útjára a versenyt. Személyes értesítéseken keresztül is 67 iskola 574 tanulója jelentkezett, s közülük 68-an kerültek az országos döntőbe. A következő tanévben a versenyt már hivatalosan is meghirdették, ennek hatására a létszám kb. 50 százalékkal nőtt (101 iskola, 893 versenyző). A verseny közben merült fel, hogy nagyon sok 3-5. osztályos tanuló is részt vett az első fordulóban, s ott igen jó eredményt értek el, de a többségük – koránál fogva – nem volt versenyképes a 8. osztályosokkal. Ezért verseny közben az Országos Versenybizottság úgy döntött, hogy a döntőt két korcsoportra bontja.

Az 1999/2000-es tanévben emiatt már eleve két kategóriában rendezték a versenyt. A versenyzői további létszám növekedése miatt az Országos Versenybizottság a 2001/2002-es tanévben a versenyt három, a 2002/2003-as tanévben pedig négy kategóriában és három fordulóban hirdette meg.

A Logo versenyhez megjelentek már példatárak<sup>1</sup>, tankönyvek, de eddig nem született olyan tananyag, ami a versenyre felkészülést támogatná.

Ez a tankönyv a Logo versenyen gyakran előforduló témaköröket tekinti át, az egyes témaköröket fokozatosan építi fel. Felépítése miatt a Logo nyelvet már ismerőknek szóló tehetséggondozó szakköri anyagnak javasoljuk. Tartalma alapján a tehetséggondozó szakkört tartó tanároknak szól, emiatt tananyag az ELTE informatika szakos tanárképzésében.

A tárgyalt témakörök:

- Sokszögek, csillagok
- Körök, körívek rajzolása
- Variációk zászlók rajzolására
- Térkitöltés forgatással
- Sorminták
- Mozaik – sorminták egymás fölé
- Ásványok, molekulák, kristályok
- Rekurzió
- Mozaik rekurzívan
- Optikai csalódások
- Variációk fa rajzolásra
- Fraktálok
- Számításokkal vezérelt rajzolás
- A környezet érzékelésével vezérelt rajzolás
- Szöveggel, listákkal vezérelt rajzolás
- Szövegmanipuláció

---

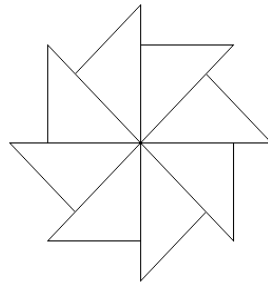
<sup>1</sup> <http://logo.inf.elte.hu/>

## Sokszögek, csillagok

Ebben a leckében sokszög és csillag alakzatok rajzolásával oldunk meg feladatokat. Kezdjük a sokszögekkel, azon belül is a háromszögek rajzolásával.

### Derékszögű háromszögekből álló alakzatok

Alapfeladat: Készíts derékszögű, egyenlőszárú háromszöget (derék), majd olyan eljárást (deréka) amelyek derékszögű háromszögekből a következő ábrát tudja kirakni:



Paraméterrel lehessen megadni a rajzolt háromszög befogójának hosszát!

Megjegyzés: A derék háromszög hosszabbik oldalát így számolhatod: befogó \* gyök(2).

A derékszögű háromszög rajzolásakor először megrajzoljuk a két befogót, majd az átfogót.

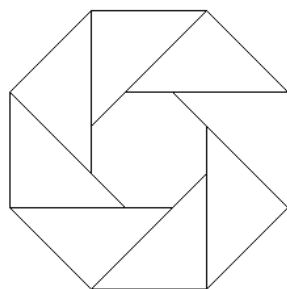
```
eljárás derék :befogó
  előre :befogó jobbra 90 előre :befogó jobbra 135
  előre :befogó*gyök 2 jobbra 135
vége
```

Az ábrát úgy kapjuk, hogy 45 fokként elfordulunk, és kirajzoljuk a derékszögű háromszöget.

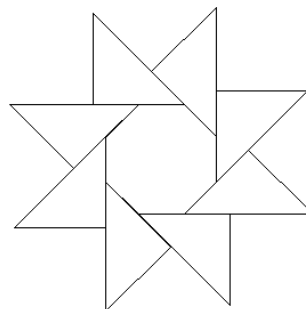
```
eljárás deréka :befogó
  ismétlés 8 [derék :befogó jobbra 45]
vége
```

### Elemi feladatvariációk

A fenti deréka eljárás felhasználásával oldjuk meg az alábbi feladatvariációkat:



1. variáció  
derékb



2. variáció  
derékc

### 1. variáció: derékb

Ezen ábrát úgy kapjuk, hogy nem csak 45 fokként fordulunk, hanem előre is lépünk a teknőccel átfogó-befogó távolságnyt.

```
eljárás derékb :befogó
  ismétlés 8 [derék :befogó jobbra 45 előre :befogó*(gyök 2)-:befogó]
vége
```

### 2. variáció: derékc

Ebben a variációban szintén 45 fok lesz az elfordulás szöge, azonban ebben az esetben minden háromszög kirajzolása után hátra lépünk a teknőccel a befogó felének megfelelő távolsággal.

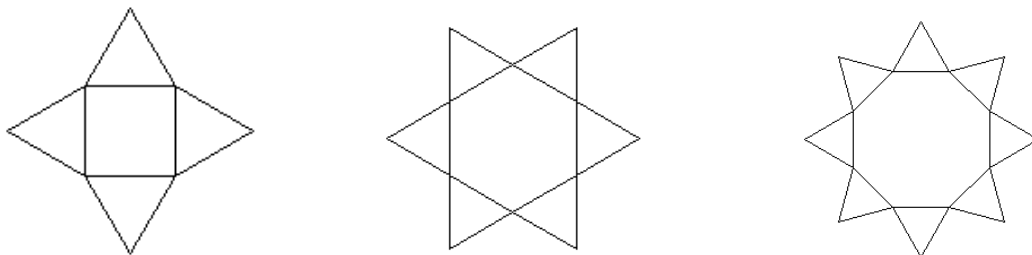
```
eljárás derékc :befogó
  ismétlés 8 [derék :befogó jobbra 45 hátra :befogó/2]
vége
```

## Szabályos háromszögeket tartalmazó alakzatok

Először tekintsük át, hogy a szabályos sokszögek és szabályos háromszögek kombinálásával milyen alakzatok, csillagformák állíthatók elő.

### Háromszögek illesztése szabályos sokszögek oldalára

Alapfeladat: Az alábbi 3 ábrát egyetlen Logo eljárás rajzolta, különböző paraméterekkel. Készíts Logo eljárást, amely ugyanezt tudja!



Láthatjuk, hogy ezekben az esetekben szabályos sokszögeket rajzolunk, ahol a sokszög oldalain szabályos háromszögeket helyezünk el. A szabályos háromszögek oldalhossza azonos, belső szögeik pedig 60 fokosak. Ebben az esetben balra fordulunk, vagyis a háromszög a teknőc pozíciójához képes balra fog esni. Erre utal az eljárás nevében a b betű.

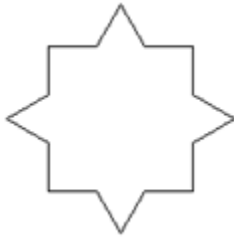
```
eljárás szabháromszögb :h
  ismétlés 3 [előre :h balra 120]
vége

eljárás sokhszog :n :h
  ismétlés :n [szabháromszögb :h előre :h jobbra 360/:n]
vége
```

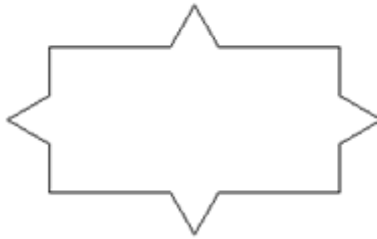
### Háromszögek illesztése szabályos sokszögek oldalára, a körvonal megrajzolásával

Szabályos háromszögeket tartalmazó ábrákat úgy is rajzolhatunk, ha az ábra körvonalát rajzoljuk meg. Erre mutat példát a következő feladat:

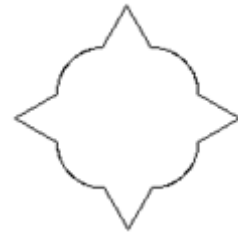
Írd meg a csillag1, csillag2, csillag3 eljárásokat, amelyek ilyen csillagokat rajzolnak!



csillag1 20



csillag2 20



csillag3 20

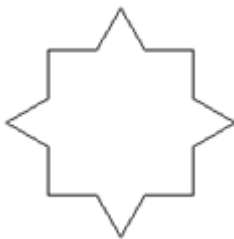
```
eljárás csillag1 :h
  ismétlés 4 [előre :h balra 60 előre :h jobbra 120
             előre :h balra 60 előre :h jobbra 90]
vége
```

```
eljárás csillag2 :h
  ismétlés 2 [előre :h balra 60 előre :h jobbra 120
             előre :h balra 60 előre :h jobbra 90 előre :h*2,5
             balra 60 előre :h jobbra 120 előre :h balra 60
             előre :h*2,5 jobbra 90]
vége
```

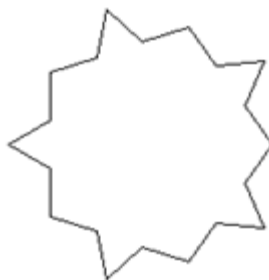
```
eljárás csillag3 :h
  ismétlés 4 [ismétlés 90 [előre 0,5 jobbra 1] balra 60 előre :h
              jobbra 120 előre :h balra 60]
vége
```

### Variációk

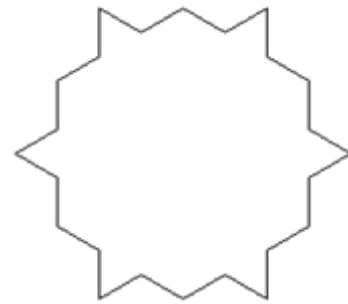
Nézzük, hogy a csillag1 ábrának milyen variációit készíthetnénk el! Karácsonyi csillagokat rajzolhatunk úgy, hogy szabályos sokszögek oldalaira V-mintát helyezünk. Írd meg a csillagv :hossz :csúcs eljárást ilyen csillagok megrajzolására!



csillagv 60 4



csillagv 60 5



csillagv 60 6

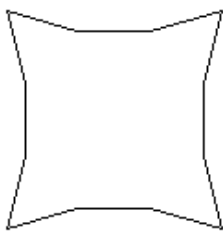
A feladatot érdemes ciklussal megvalósítani, ahol egy lépésben oldalt rajzolunk meg és gondoskodunk a megfelelő elfordulásról is.

```
eljárás csillagv :hossz :n
  ismétlés :n [előre :hossz/3 balra 60 előre :hossz/3 jobbra 120
              előre :hossz/3 balra 60 előre :hossz/3 jobbra 360/:n]
vége
```

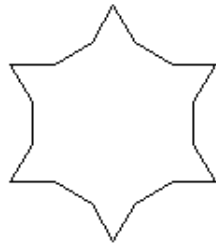
### Szabályos háromszög illesztése szabályos sokszögek csúcsaira

Azt is megtehetjük, hogy a szabályos háromszögeket nem a szabályos sokszögek oldalaira, hanem a csúcsaira ültetjük.

Írd meg a csúcsos :átmérő :csúcs eljárást, amely karácsonyi díszeket rajzol! Az átmérőn az alakzatba írható kör átmérőjét, csúcson pedig a sokszög csúcsainak számát értjük. A háromszögek oldalhosszúsága és az összekötő szakaszok hosszúsága azonos.



csúcsos 40 4



csúcsos 40 6



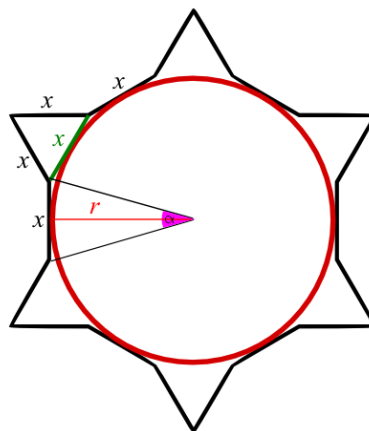
csúcsos 40 9



csúcsos 40 12

Ezt a feladatot az előbbi feladatra is vissza lehetne vezetni a háromszög rajzoló eljárás módosításával, azonban most egy más jellegű megoldást ismertetünk. A feladat nehézsége abban rejlik, hogy ki kell számítanunk a háromszögek oldalhosszát és a fordulási szögeket. Ehhez magasabb matematikai tudás szükséges, mivel ismerni kell a szögfüggvényeket.

Kezdjük az oldal hosszának kiszámításával! Magyarozatképpen nézzük az alábbi sematikus ábrát, amit akkor kapunk, ha a csúcsok számának hatot adunk meg! Mivel a csúcsokra ültetett szabályos háromszögek oldalhossza és azokat összekötő szakaszok hossza ugyanaz (az ábrán  $x$ -el jelölve), ebben az esetben egy szabályos 12 szög (vagyis a paraméterül adott csúcsszám kétszeresének megfelelő) szabályos sokszöget kapunk.



Az ábrán jelölt  $\alpha$  szöget úgy kapjuk, hogy a teljes szöget (360 fok) elosztjuk a paraméterként megadott csúcsszám kétszeresével. Az  $x$  szakasz hosszát a kör sugarából ( $r$ ) és az  $\alpha$  szögből ki tudjuk számítani.

$$\frac{x}{2} = \operatorname{tg} \alpha$$

A fenti képletből azt kapjuk, hogy:  $x=2r \operatorname{tg} \alpha = d \operatorname{tg} 180/\text{csúcsok száma}$ .

(A  $d$  a megadott átmérőt jelenti.)

Ha már ismerjük az  $x$  szakasz hosszát, akkor a fordulási szögek kiszámítása után már meg tudjuk rajzolni az ábrát. A szabályos sokszög megrajzolásakor a teknőccel  $360/\text{csúcsok száma}$  szöggel kell

elfordulnunk. az oldal megrajzolása után. A megadott csúcyszámot viszont ebben az esetben kétszeresen kell vennünk, a korábban részletezett okok miatt. A szabályos háromszög belső szögei 60 fokosak , így ezek alapján már megrajzolható az ábra:

```

eljárás csúcsos :átmérő :csúcs
  ismétlés :csúcs [előre :átmérő*tg (180/:csúcs) jobbra 360/(2*:csúcs)
    balra 60 előre :átmérő*tg (180/:csúcs) jobbra 120
    előre :átmérő*tg (180/:csúcs)
    balra 60 jobbra 360/(2*:csúcs)]

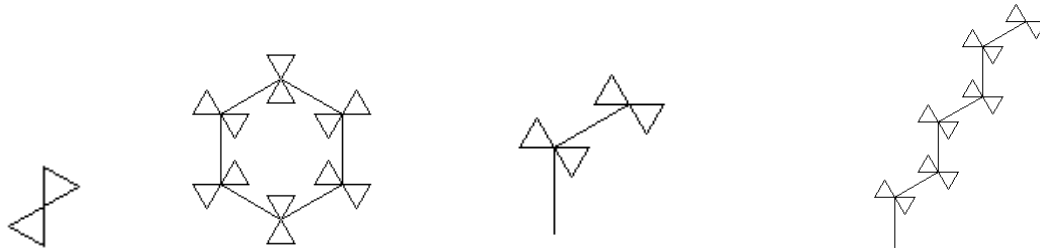
```

vége

### Háromszögek szembefordítása, masnik rajzolása

Egy masnit úgy kell rajzolni, hogy két szabályos háromszöget szembefordítunk egymással. A masnikat drótvázzal kétféleképpen köthetjük össze. Egyik esetben egy hatszög oldalait alkotják az összekötő drótszakaszok, a másikban pedig egymás után fűzzük őket, hol balra, hol jobbra fordulva.

Alapfeladat: Készíts eljárásokat (masni :méret, hatszög :méret :dróthossz, duplamasni :méret :dróthossz, masnisor :db :méret :dróthossz), ahol :méret a masni háromszögeinek oldalhossza, :dróthossz a masnikat összekötő drót hossza, :db pedig a masnisorban levő duplamasnik száma.



```

masni 20 hatszög 10 30
duplamasni 10 30
masnisor 3 10 30

```

Ebben az esetben a háromszöget érdemes jobbra fordulva megrajzolni, így picit módosul a korábbi háromszög rajzoló eljárásunk, melyben a j betű utal arra, hogy jobbra fordulva rajzoljuk meg a háromszöget.

```

eljárás szabháromszögj :h
  ismétlés 3 [előre :h jobbra 120]
vége

```

A masni két egymással szembe fordított szabályos háromszöggé lesz megrajzolva.

```

eljárás masni :hossz
  ismétlés 2 [szabháromszögj :hossz jobbra 180]
vége

```

A hatszög megrajolásánál minden csúcsban megfelelő szöggel (90) való elfordulás után rajzoljuk meg a masnikat.

```

eljárás hatszög :hossz :dróthossz
  ismétlés 6 [előre :dróthossz jobbra 90 masni :hossz balra 30]
vége

```

A dupla masni megrajolásakor dróthossznyi távolságot kell előre menni, majd megfelelő szögben megrajzolni a masnit, elfordulni, majd más paraméterekkel megismételni a rajzolást.

```
eljárás duplamasni :méret :dróthossz
  előre :dróthossz jobbra 90 masni :méret balra 30
  előre :dróthossz jobbra 30 masni :méret balra 90
vége
```

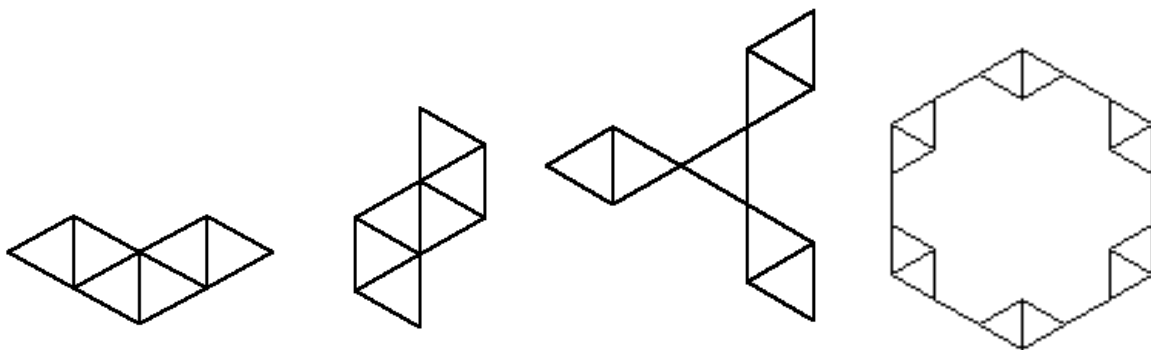
A masnisor adott darabszámú dupla masni megrajzolásával születik:

```
eljárás masnisor :db :méret :dróthossz
  ismétlés :db [duplamasni :méret :dróthossz]
vége
```

## Szabályos háromszögekből álló jelvények rajzolása

### 1. variáció

A nyári táborban gyerekek háromszögre alapozott jelvényeket terveznek. Írjuk meg az alábbi jelvényeket rajzoló eljárásokat (jel1,jel2,jel3,jel4)! Az eljárásoknak lehessen megadni az oldalhosszt paraméterként!



jel1 50

jel2 50

jel3 50

jel4 50

Az alap eljárást úgy készítjük, hogy rajzolunk balra fordulva és jobbra fordulva is egy szabályos háromszöget a megadott oldalhosszal.

```
eljárás jelalap :h
  szabháromszögj :h szabháromszögb :h
vége
```

Az első jelvényt három darab alapjelből állítjuk össze, majd gondoskodunk arról, hogy az eredeti helyzetbe kerüljön a teknőc.

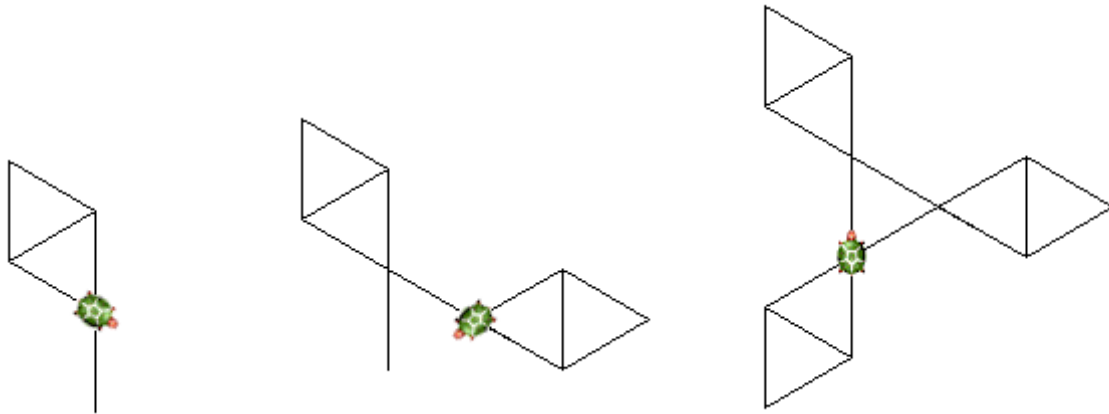
```
eljárás jel1 :h
  jelalap :h jobbra 60 előre :h balra 60
  jelalap :h jobbra 60 hátra :h balra 120 előre :h jobbra 60
  jelalap :h
  tollatfel balra 60 hátra :h jobbra 60 tollatle
vége
```

A második jelvény szintén három darab alapjelből áll össze, de a teknőcök az előzőhöz képest más szöggel fordulnak el a rajzolás során. Itt is gondoskodunk arról, hogy a végén itt is eredeti helyzetbe kerüljön a teknőc.

```
eljárás jel2 :h
  jelalap :h balra 120
  jelalap :h jobbra 120 előre :h jobbra 60
  jelalap :h
  tollatfel balra 60 hátra :h tollatle
vége
```



A harmadik jelvénynél háromszor ismétljük ugyanazt a rajzoló ciklust. Az egyes ciklusok eredményét az alábbi képeken láthatjuk:



Itt is gondoskodunk arról, hogy a végén itt is eredeti helyzetbe kerüljön a teknőc.

```

eljárás jel3 :h
  jobbra 60
  ismétlés 3 [előre :h*2 balra 120 jelalap :h
              jobbra 120 hátra :h jobbra 120]
  balra 60
vége
    
```

A negyedik jelvénynél a háromszög oldalhosszának háromszorosa lesz a hatszög oldalhossza.

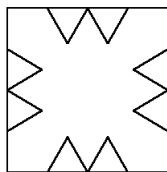
```

eljárás jel4 :h
  ismétlés 6 [előre :h*3 jobbra 120 jelalap :h balra 60]
vége
    
```

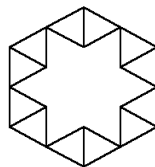
## 2. variáció

Jelvényeket úgy is rajzolhatunk, ha a szabályos háromszögeket nem egymással szembefordítva rajzoljuk, hanem egymás mellé, mintha fogak lennének.

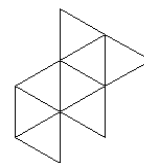
A nyári táborban gyerekek háromszögre alapozott jelvényeket terveznek. Írd meg az alábbi jelvényeket rajzoló eljárásokat! Az átadott paraméter a szabályos háromszög oldalhosszát jelentsé!



jelfog1 50



jelfog2 50



jelfog3 50

A fogat két jobb oldalra rajzolt háromszöggel alkotjuk meg.

```

eljárás fog :h
  ismétlés 2 [szabháromszögj :h előre :h]
vége
    
```

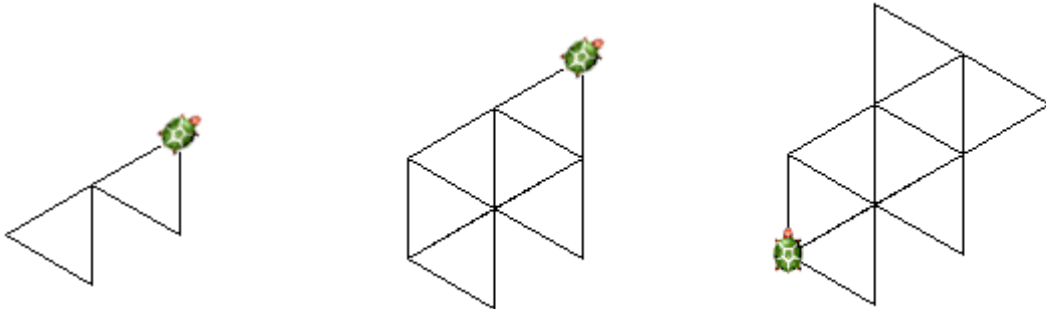
Az első jelvényt úgy rajzoljuk meg, hogy a négyzet oldalhossza a háromszög oldalhosszának négyszerese legyen, ehhez először előrelépünk oldalhossznyit, megrajzoljuk a fogat (ami két oldalhossznyi), majd újra előrelépünk oldalhossznyit és jobbra fordulunk 90 fokkal.

```
eljárás jelfog1 :h
  ismétlés 4 [előre :h fog :h előre :h jobbra 90]
vége
```

A második esetben a hatszög oldalai pont egy megrajzolt fogból állnak.

```
eljárás jelfog2 :h
  ismétlés 6 [fog :h jobbra 60]
vége
```

A harmadik jelvény megrajzolása során többféle módszert lehetne használni, mi az alábbi sorrendben rajzoltuk meg a jelvényt:

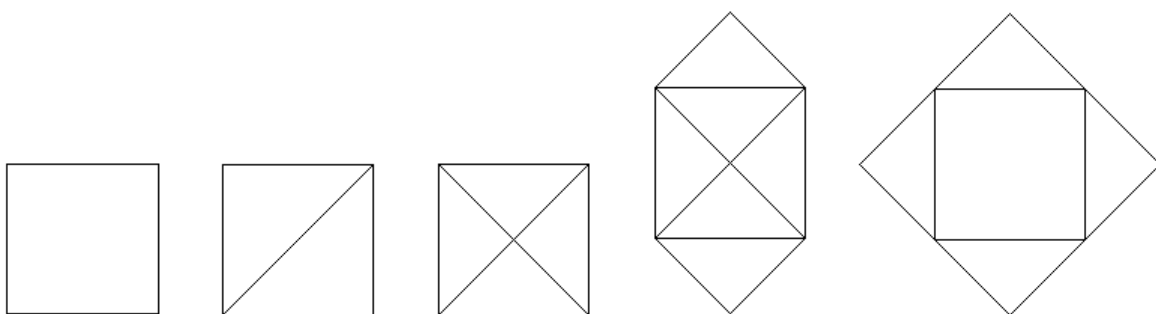


```
eljárás jelfog3 :h
  jobbra 60 fog :h hátra 2*:h balra 60 előre :h jobbra 60
  fog :h jobbra 60 hátra :h fog :h
  tollatfel balra 60 hátra 3*:h balra 60 tollatle
vége
```

## Négyszögek rajzolása

### Négyzetből alkotott alakzatok készítése

Írj Logo eljárásokat (négyzet1 :oldal, ..., négyzet5 :oldal) az alábbi négyzetek rajzolására, ahol :oldal a négyzet oldalhossza! A négyzet átlójának hossza az oldalhossz gyök(2)-szöröse!



négyzet1 50    négyzet2 50    négyzet3 50    négyzet4 50    négyzet5 50

```
eljárás négyzet1 :h
  ismétlés 4 [előre :h jobbra 90]
vége
```

```
eljárás négyzet2 :h
  ismétlés 4 [előre :h jobbra 90]
  jobbra 45 előre :h*gyök 2 hátra :h*gyök 2 balra 45
vége
```

```

eljárás négyzet3 :h
  ismétlés 4 [előre :h jobbra 135 előre :h/2*gyök 2
             hátra :h/2*gyök 2 balra 45]

```

vége

```

eljárás négyzet4 :h
  ismétlés 2 [jobbra 45 előre :h/2*gyök 2 balra 90
             előre :h/2*gyök 2 jobbra 45 hátra :h előre :h
             jobbra 45 előre :h/2*gyök 2 jobbra 90
             előre :h/2*gyök 2 balra 45 hátra :h előre :h
             jobbra 90]

```

vége

```

eljárás négyzet5 :h
  ismétlés 4 [balra 45 előre :h/2*gyök 2 jobbra 90
             előre :h/2*gyök 2 balra 45 hátra :h előre :h jobbra 90]

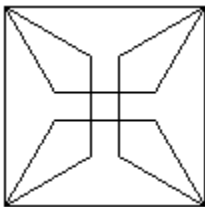
```

vége

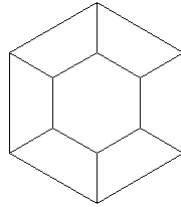
### Trapézból alkotott alakzatok rajzolása

Az alábbi 3 ábrát egyetlen Logo eljárás rajzolta, különböző paraméterekkel.

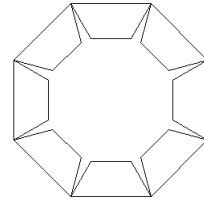
Készíts egy trapézok nevű eljárást, amely ugyanezt tudja!



trapézok 4 100



trapézok 6 50



trapézok 8 50

```

eljárás trapéz :h
  előre :h jobbra 120 előre :h/2 jobbra 60
  előre :h/2 jobbra 60 előre :h/2 jobbra 120
vége

```

```

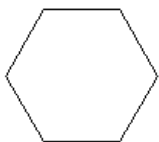
eljárás trapézok :n :h
  ismétlés :n [trapéz :h előre :h jobbra 360/:n]
vége

```

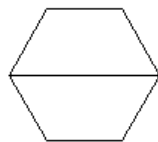
### Hatszögvariációk

#### 1. feladat

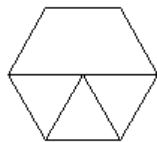
Írj Logo eljárásokat (hatszög1 :oldal, ..., hatszög5 :oldal) az alábbi hatszögek rajzolására, ahol :oldal a hatszög oldalhossza!



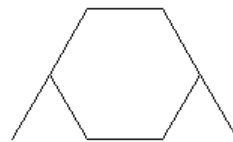
hatszög1 50



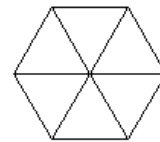
hatszög2 50



hatszög3 50



hatszög4 50



hatszög5 50

```

eljárás hatszög1 :h
  balra 30 ismétlés 6 [előre :h jobbra 60] jobbra 30
vége

```

```
eljárás hatszög2 :h
  balra 30 ismétlés 6 [előre :h jobbra 60]
  előre :h jobbra 120 előre 2*:h hátra 2*:h balra 120 hátra :h
  jobbra 30
vége
```

```
eljárás hatszög3 :h
  balra 30 szabháromszögj :h előre :h jobbra 60
  ismétlés 3 [előre :h jobbra 60]
  ismétlés 2 [szabháromszögj :h előre :h jobbra 60]
  jobbra 30
vége
```

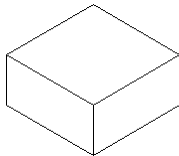
```
eljárás hatszög4 :h
  balra 30 előre :h jobbra 60 hátra :h előre :h
  ismétlés 3 [előre :h jobbra 60]
  balra 60 előre :h hátra :h jobbra 60
  ismétlés 2 [előre :h jobbra 60]
  jobbra 30
vége
```

```
eljárás hatszög5 :h
  balra 30
  ismétlés 6 [szabháromszögj :h előre :h jobbra 60]
  jobbra 30
vége
```

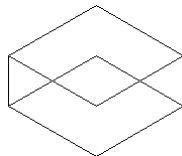
```
eljárás szabháromszögj :h
  ismétlés 3 [előre :h jobbra 120]
vége
```

## 2. feladat

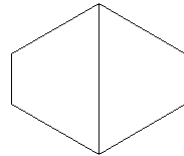
Készíts eljárásokat HATA :oldal :oldal2, HATB :oldal :oldal2, HATC :oldal :oldal2, HATD :oldal :oldal2, HATE :oldal :oldal2, amelyek az alábbi hatszögféleségeket rajzolják:



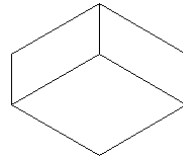
HATA 50 100



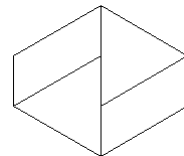
HATB 50 100



HATC 50 100



HATD 50 100



HATE 50 100

```
eljárás HATA :g :h
  ismétlés 2 [előre :g jobbra 60 előre :h jobbra 60 előre :h jobbra 60]
  jobbra 120 előre :h balra 120 előre :g balra 60 előre :h
  hátra :h jobbra 120 előre :h hátra :h balra 60 hátra :g
  balra 60 előre :h jobbra 60
vége
```

```
eljárás HATB :g :h
  ismétlés 2 [előre :g jobbra 60 előre :h jobbra 60 előre :h jobbra 60]
  ismétlés 2 [ismétlés 2 [jobbra 60 előre :h] balra 120 előre :g
  balra 180]
vége
```

```
eljárás HATC :g :h
  ismétlés 2 [előre :g jobbra 60 előre :h jobbra 60 előre :h jobbra 60]
  balra 60 hátra :h jobbra 60 előre :h+:g hátra :h+:g balra 60
  előre :h jobbra 60
vége
```

```

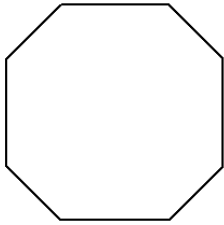
eljárás HATD :g :h
  ismétlés 2 [előre :g jobbra 60 előre :h jobbra 60 előre :h jobbra 60]
  jobbra 60 előre :h balra 60 előre :g hátra :g jobbra 120
  előre :h hátra :h balra 60 hátra :h balra 60
vége
  
```

```

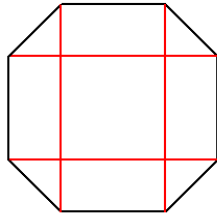
eljárás HATE :g :h
  ismétlés 2 [előre :g jobbra 60 előre :h jobbra 60 előre :h jobbra 60]
  jobbra 60 előre :h balra 60 előre :g hátra :g+:h előre :g
  jobbra 60 előre :h hátra :h balra 60 előre :h-:g jobbra 60
  hátra :h balra 60
vége
  
```

## Nyolcszögek rajzolása

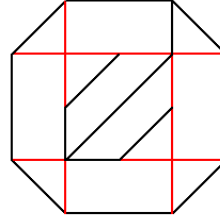
Írj Logo eljárásokat (nyolcszög1 :oldal, ... nyolcszög4 :oldal) az alábbi nyolcszögek rajzolására, ahol :oldal a nyolcszög oldalhossza! A nyolcszöget úgy kapjuk, hogy egy négyzet 4 csúcsát levágjuk.



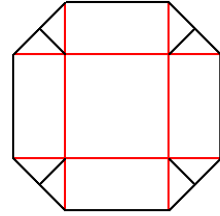
nyolcszög1 100



nyolcszög2 100



nyolcszög3 100



nyolcszög4 100

```

eljárás nyolcszög1 :oldal
  ismétlés 4 [előre :oldal/2 jobbra 45 előre :oldal/4*gyök 2 jobbra 45]
vége
  
```

```

eljárás nyolcszög2 :oldal
  ismétlés 4 [előre :oldal/2 jobbra 45 előre :oldal/4*gyök 2 jobbra 45]
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 előre :oldal/2
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90
  jobbra 45 előre :oldal/4*gyök 2 jobbra 45
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 b 90 előre :oldal/2
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90
vége
  
```

```

eljárás nyolcszög3 :oldal
  ismétlés 4 [előre :oldal/2 jobbra 45 előre :oldal/4*gyök 2 jobbra 45]
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 előre :oldal/2
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 jobbra 45 előre :oldal/4*gyök 2 jobbra 45
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 előre :oldal/2 tollszín! 12
  jobbra 90 előre :oldal hátra :oldal tollszín! 0
  előre :oldal/4 jobbra 45 előre :oldal/2*gyök 2
  jobbra 135 előre :oldal/4 jobbra 45 előre :oldal/4*gyök 2
  hátra :oldal/4*gyök 2 balra 45 hátra :oldal/4
  jobbra 90 előre :oldal/4 balra 45 előre :oldal/4*gyök 2
vége
  
```

```

eljárás nyolcszög4 :oldal
  ismétlés 4 [előre :oldal/2 jobbra 45 előre :oldal/4*gyök 2 jobbra 45]
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 előre :oldal/2
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 jobbra 45 előre :oldal/4*gyök 2 jobbra 45
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 90 előre :oldal/2
  tollszín! 12 jobbra 90 előre :oldal hátra :oldal
  tollszín! 0 balra 45
  ismétlés 4 [előre :oldal/8*gyök 2 jobbra 90
              előre :oldal/8*gyök 2 hátra :oldal/8*gyök 2
              balra 90 előre :oldal/8*gyök 2 jobbra 45
              előre :oldal/2 jobbra 45]

```

vége

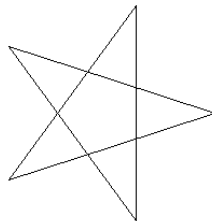
## Törött sokszög rajzolása

Törött vonalú sokszöget úgy rajzolunk, hogy a sokszög oldalait helyettesítjük két szakasszal, amelyek az eredeti oldallal  $F$  fokos szöget zárnak be.

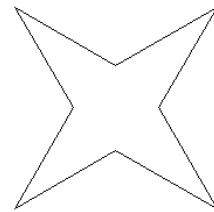
Készíts Logo eljárást (Tsokszög :h :f :s), amely olyan sokszöget rajzol, amelynél az oldalakat helyettesítő szakaszpárok hossza :h, ezek :f fokos szöget zárnak be az eredeti sokszöggel, s a sokszög csúcaiban :s fokot kell fordulni! Egy oldalt az Oldal :h :f eljárás rajzoljon! A sokszög rajzolása akkor fejeződjön be, ha a teknőc visszatér a rajzolás előtti állapotába!



oldal 100 30



tsokszög 100 0 144



tsokszög 100 30 90

```

eljárás oldal :h :f
  balra :f előre :h jobbra 2 * :f előre :h balra :f
  vége

eljárás tsokszög :h :f :s
  törtsok :h :f :s 0
  vége

eljárás törtsok :h :f :s :szög
  oldal :h :f balra :s
  ha maradék :s+:szög 360 > 0 [törtsok :h :f :s :s+:szög]
  vége

```

## Csillagok rajzolása

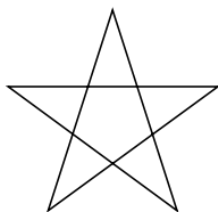
Az előbbi feladatban, adott paraméter megadása után csillag alakzatot kaptunk. Nézzük meg, hogy milyen lehetőségeink vannak még csillag alakzatok rajolására!

### Szabályos csillagsokszögek

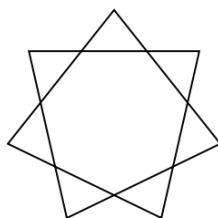
Mielőtt belekezdünk a feladat megoldásába, tekintsük át a kapcsolódó alapfogalmakat. Csillagsokszögnek azon síkbeli zárt töröttvonal alakzatot nevezzük, ami metszi saját magát. A csillagsokszögekben belül is megkülönböztetjük a szabályos csillagsokszögeket, ahol a csúcsokban mért szögek megegyeznek.

Csillagsokszöget kapunk, ha egy szabályos sokszög csúcsait összekötjük a nem szomszédos csúcsokkal, akár úgy, hogy bizonyos szabály szerint nem az összes nem-szomszédjával kötjük össze. A csillagsokszögek jelölésére a következő szimbólumot használjuk:  $\{n/k\}$ , ahol  $n$  jelöli, hogy hány csúcsa van a szabályos sokszögnek,  $k$  pedig azt jelöli, hogy hányadik szomszédjával van összekötve egy csúcs<sup>2</sup>.

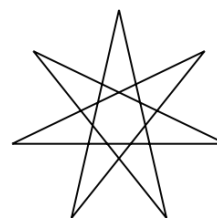
Nézzünk néhány szabályos csillagsokszöget, a fenti jelölés alkalmazásával<sup>3</sup>.



$\{5/2\}$



$\{7/2\}$



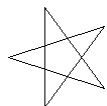
$\{7/3\}$

Szabályos csillagsokszög alakzatokat úgy rajzolhatunk a teknőccel, hogy előrelépünk, jobbra fordulunk megadott szöggel, és ezt annyiszor ismételjük, ahány csúcsa van a csillagnak. De mekkora szöggel kell elfordulnunk?

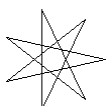
A Teljes Teknőc Tétel kimondja, hogy a zárt síkbeli alakzatok rajzolásakor, amennyiben a teknőc visszatér kiindulási állapotába, akkor a fordulatok összege  $360^\circ$  vagy annak többszöröse. Ha ezt a szöveget elosztjuk a csúcsok számával, akkor megkapjuk, hogy mennyit kell elfordulnia a teknőcnek a csillag rajzolásakor. Persze ki kell kísérleteznünk, hogy az adott ábra rajzolásakor a  $360$  fok melyik többszörösét kell leosztanunk a csúcsok számával.

Nézzünk a következő alapfeladatot!

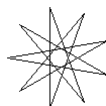
Alapfeladat: Készíts Logo eljárásokat (F1A, F2A, F3A, F4A), amelyek az alábbi ábrákat rajzolják!



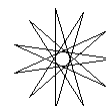
F1A



F2A

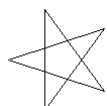


F3A



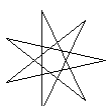
F4A

Mielőtt nekilátnánk a feladat megoldásának, nézzük meg, hogy a fenti ábrák az  $\{n/k\}$  szimbólum segítségével hogyan írhatóak le ( $n$ : csúcsok száma,  $k$ : hányadik szomszédjával van összekötve egy csúcs).



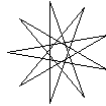
F1A

$\{5/2\}$



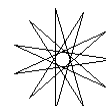
F2A

$\{7/3\}$



F3A

$\{9/4\}$



F4A

$\{11/5\}$

Ezek után rátérhetünk a megoldásra. A szabályos csillagsokszögek megrajzolásakor tehát annyiszor kell ismételni az előrelépést és a jobbra fordulást, amennyi csúcsa ( $n$ ) van a csillagnak. Az elfordulás

<sup>2</sup> [https://www.cs.elte.hu/blobs/diplomamunkak/bsc\\_mattan/2012/kovacs\\_maria.pdf](https://www.cs.elte.hu/blobs/diplomamunkak/bsc_mattan/2012/kovacs_maria.pdf)

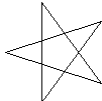
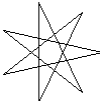

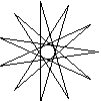
<sup>3</sup> <http://hu.wikipedia.org/wiki/Csillagsoksz%C3%B6g>

szögét úgy kapjuk, hogy a paraméterként megadott szöget, ami a 360 többszöröse lehet, elosztjuk a csúcsok számával. Vagyis az eljárásunk a következő lehet:

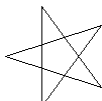
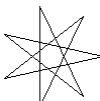
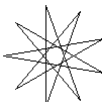
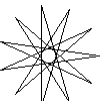
```
eljárás csillag :n :szög :hossz
  ismétlés :n [előre :hossz jobbra :szög/:n]
vége
```

Próbáljuk kikísérletezni, hogy milyen paraméterek megadásával tudjuk megrajzolni a kívánt ábrákat! A csúcsok számát nyilván ismerjük, a hossz tetszőleges, ezért a megadott szöggel kell kísérleteznünk, vagyis meg kell találnunk, hogy a 360 fok mely többszörösével kapjuk meg az ábrát.

Azt kapjuk, hogy az ábrák rajzolásához az alábbi paramétereket kell megadni:

			
csillag 5	csillag 7	csillag 9	csillag 11
360*2 100	360*3 100	360*4 100	360*5 100

Ha a fenti sort kiegészítjük azzal, hogy az {n/k} jelöléssel hogyan írható le az alakzat, fontos felismerést tehetünk.

			
csillag 5	csillag 7	csillag 9	csillag 11
360* <b>2</b> 100	360* <b>3</b> 100	360* <b>4</b> 100	360* <b>5</b> 100
{5/ <b>2</b> }	{7/ <b>3</b> }	{9/ <b>4</b> }	{11/ <b>5</b> }

Szépen látszik, hogy annyszor kell megszorozni a 360 fokot a paraméter átadásnál, ahányadik szomszédal vannak a csúcsok összekötve.

Ez a szabály természetesen matematikailag is belátható. Matematikusok bebizonyították, hogy a szabályos csillagsokszögek csúcsokban mért belső szöge kiszámítható az alábbi képlettel:

$$180(n-2k)/n$$

Mivel ez a belső szög, a teknőcnek nem ennyit kell elfordulnia rajzoláskor, hanem a 180 fokból ki kell vonni ezt a szöget, vagyis az elfordulás mértéke:  $180-180(n-2k)/n$ .

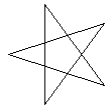
Ha azt akarjuk megtudni, hogy összesen hány fokot fordult a teknőc, akkor a fenti képletet meg kell szoroznunk a csúcsok számával:  $(180-180(n-2k)/n)*n=180n-180(n-2k)=360k$

Ez alapján tovább egyszerűsíthetjük a szabályos csillagsokszög eljárásunkat. Paraméterül ne a szöget, hanem az n és k értéket, valamint a hosszt adjuk meg!

```
eljárás szabcsillag :n :k :hossz
  ismétlés :n [előre :hossz jobbra 360*:k/:n]
vége
```

Ezzel az eljárással tehát az alábbi módokon rajzolhatjuk meg az alakzatokat.





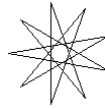
szabcsillagsokszog  
5 2 100

{5/2}



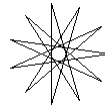
szabcsillagsokszog  
7 3 100

{7/3}



szabcsillagsokszog  
9 4 100

{9/4}



szabcsillagsokszog  
11 5 100

{11/5}

Nézzük meg azt az esetet is, amikor a szabályos csillagsokszög körvonalát kell megrajzolnunk!

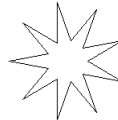
Alapfeladat: Készíts Logo eljárásokat (F1B, F2B, F3B, F4B), amelyek az alábbi ábrákat rajzolják!



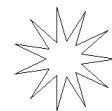
F1B



F2B



F3B



F4B

Nyilvánvaló, hogy most a teknőccel nem elég csak egy irányba fordulni, hanem változtatni kell az elfordulás szögét, vagyis az oldal megrajzolása után balra kell fordulni, majd az újabb oldalrajzolás után jobbra kell fordulni. Ezt annyiszor kell ismételni, ahány csúcsa van a csillagnak. A jobbra fordulás szöge ugyanaz lesz, mint az előző feladatban ( $360 \cdot k / n$ ), de mi a helyzet a balra fordulás szögével? A balra fordulás szöge  $((360 \cdot k) - 360) / n$  lesz.

```
eljárás uresszabcsillag :n :k :hossz
  ismétlés :n [előre :hossz balra ((360*:k)-360)/:n
               előre :hossz jobbra 360*:k/:n]
```

vége

Ajánlott linkek, források:

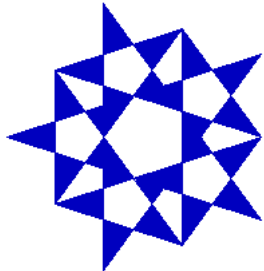
- <http://www.algebra.com/algebra/homework/Polygons/Polygons.faq.question.225075.html>
- <http://www.komal.hu/cikkek/2003-11/csillag.h.shtml>
- <http://mathworld.wolfram.com/StarPolygon.html>
- <http://hu.wikipedia.org/wiki/Csillagsoksz%C3%B6g>
- [http://en.wikipedia.org/wiki/Star\\_polygon](http://en.wikipedia.org/wiki/Star_polygon)
- <http://getmathhelp.jimdo.com/geometry/star-shaped-polygon/>
- <http://www.mathsisfun.com/geometry/interior-angles-polygons.html>
- <http://donsteward.blogspot.hu/2011/05/star-polygons.html>

## Csillagokból álló komplex ábrák

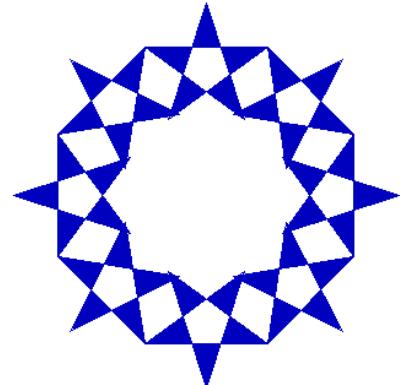
Alapfeladat: Készíts Logo eljárást színezett ágú ötágú csillag rajzolására (ötágúcsillag :h), ahol :h a csillag oldalhossza! Alkoss csillagokból :n oldalú szabályos sokszöget (csillagok :n :h), ahol a csillagok a csúcaikkal érnek össze!



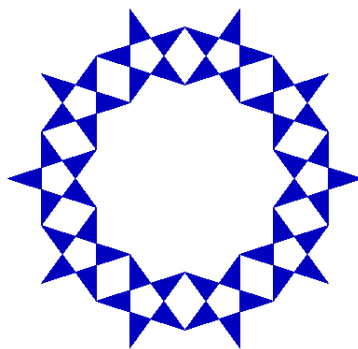
csillag 100



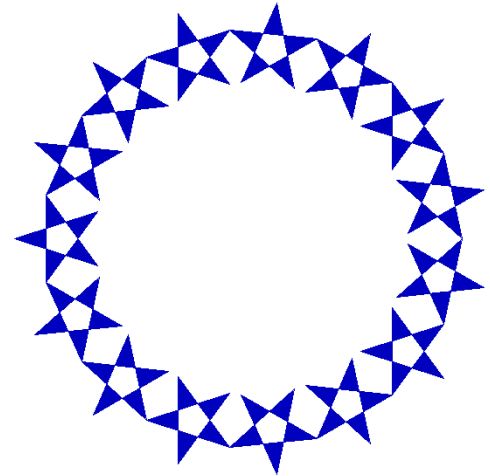
csillagok 5 100



csillagok 8 100



csillagok 10 50



csillagok 15 50

Használjuk fel a korábban megírt szabcsillag eljárásunkat! Az ötágú csillag megrajzolása után újra végig kell járni az útvonalat, és a csúcokat ki kell tölteni kék színnel-

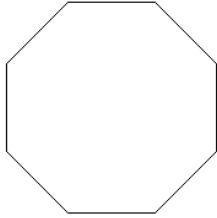
```
eljárás ötágúcsillag :hossz
  töltőszín! "sötétkék tollszín! "sötétkék
  szabcsillag 5 2 :hossz tollatfel
  ismétlés 5 [előre :hossz jobbra 360*2/5
              jobbra 10 előre :hossz/4 tölt hátra :hossz/4 balra 10]
  tollatle töltőszín! "fekete tollszín! "fekete
vége
```

A csillagok eljárásban a megadott számú csillagot kell kirajzolni, úgy hogy ezek egymáshoz a csúcsaiknál érintkezzenek, amit oldalhossznyi előre lépéssel és megfelelő szöggel való  $(360 / :n)$  elfordulással oldunk meg.

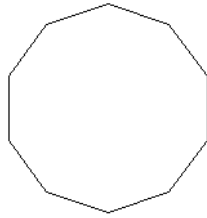
```
eljárás csillagok :n :hossz
  ismétlés :n [ötágúcsillag :hossz tollatfel előre :hossz jobbra 360/:n
              tollatle]
vége
```

## Körök, körívek rajzolása

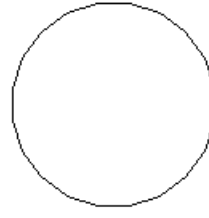
A következőkben különböző kör rajzoló módszereket mutatunk meg, amelyeket különböző jellegű körrajzoló feladatok megoldása során felhasználhatunk. A kör rajzolása során azt használjuk ki, hogy minél több oldala van egy szabályos sokszögnek, annál inkább egy körhöz hasonlít.



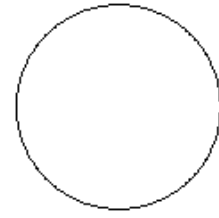
Szabályos nyolcszög



Szabályos tízsög

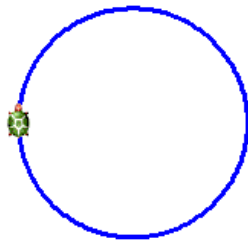


Szabályos húszszög

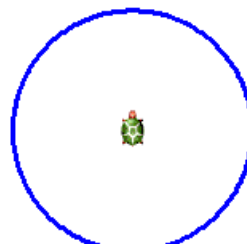


Szabályos 360 oldalú sokszög képe

A Teljes Teknőc Tétel kimondja, hogy a zárt síkbeli alakzatok rajzolásakor, amennyiben a teknőc visszatér kiindulási állapotába, akkor a fordulatok összege  $360^\circ$  vagy annak többszöröse. Ha a teknőccel 360-szor ismételtjük az egy egységgel történő előrelépést, és az  $1^\circ$ -al történő elfordulást, akkor visszatérünk a kiindulási helyzetbe, és eredményül egy kört kapunk. Ezen elvre épülve két körrajzoló eljárást is bemutatunk, amelyek abban különböznek, hogy a teknőc kiindulási pozíciója a körvonalon helyezkedik-e el, vagy a kör középpontjában.



Kör rajzolása, ha a teknőc pozíciója a körvonalon van



Kör rajzolása, ha a teknőc pozíciója a kör középpontjában van

### Körvonal rajzolása (ha a teknőc pozíciója a körvonalon van)

Alapfeladat: Készíts olyan körrajzoló eljárást, amely úgy rajzol kört, hogy a teknőc pozíciója a körvonalon helyezkedik el. Paraméterként lehessen megadni a kör sugarát, a körvonal vastagságát, színe!

Eljárásunkat `kör_körvonal` néven készítjük, jelezve, hogy a teknőc ebben az esetben a körvonalon helyezkedik el.

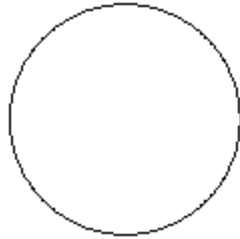
```
eljárás kör_körvonal
  ismétlés 360 [előre 1 jobbra 1]
vége
```

Az ezen eljárással rajzolt kör kerülete 360 egység, így a kör sugara kiszámítható a  $360/2\pi$  képlettel, vagyis a sugár ebben az esetben  $r \approx 57,29$  egység. Természetesen, ha nem 1 lépést tennénk előre, hanem tetszőleges lépést (amelyet paraméterrel adnánk meg), a kör kerülete, vagyis mérete is megváltozna. Paraméterként akár tört számot is megadhatunk, mint ahogy az az alábbi példákban is látható.

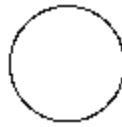
A továbbfejlesztett eljárás:

```
eljárás kör_körvonal :hossz
  ismétlés 360 [előre :hossz jobbra 1]
vége
```

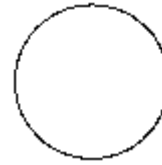
Az eljárással megrajzolt körök különböző paraméterek esetén:



kör\_alap 1



kör\_alap 0,5



kör\_alap 2/3

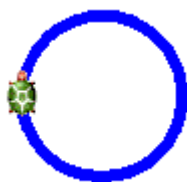
A fenti eljárásban megadott `:hossz` érték tehát a kör kerületére van hatással. Ha inkább a sugárt szeretnénk paraméterül átadni, akkor módosítanunk kell az eljárást. Ekkor a sugárból kiszámolt kerület 360-ad részével kell előre lépünk egy lépésben.

```
eljárás kör_körvonal :sugár
  ismétlés 360 [előre :sugár*2*3,14159/360 jobbra 1]
vége
```

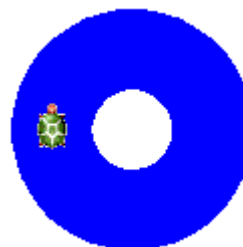
Így már adott sugarú kört fog rajzolni az eljárásunk. Fejlesszük tovább az eljárást, hogy paraméterként a körvonal vastagságát és színét is meg lehessen adni!

```
eljárás kör_körvonal :sugár :vastagság :körvonalszín
  tollvastagság! :vastagság tollszín! :körvonalszín
  ismétlés 360 [előre :sugár*2*3,14159/360 jobbra 1]
  tollvastagság! 1 tollszín! 0
vége
```

Próbáljuk ki az eljárást különböző paraméterekkel:



kör\_körvonal 40 6 "kék



kör\_körvonal 40 40 "kék

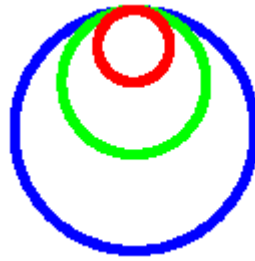
Vegyük észre, hogy a fenti két körnek ugyanaz a sugara, de a körvonal vastagságban különbség van! Vagyis a megrajzolt kör sugara a paraméterként megadott sugár és a toll vastagság felének az összege lesz.

## Elemi feladatvariációk

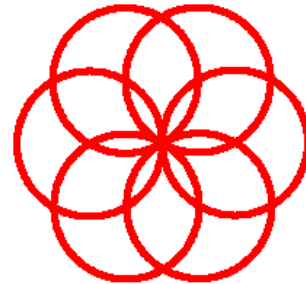
A fenti eljárás segítségével oldjuk meg az alábbi feladatvariációkat:



1. variáció  
Gumimatrac



2. variáció  
Medál



3. variáció  
Virág

### 1. variáció: Gumimatrac

A feladatot három, azonos sugarú, de eltérő vastagságú körvonal rajzolásával megoldhatjuk. A fekete körvonal mérete a sugár kétharmada, a piros mérete a sugár fele, míg a fehér körvonal mérete a sugár egytizede legyen.

```
eljárás gumimatrac :méret
  kör_körvonal :méret :méret*2/3 "fekete
  kör_körvonal :méret :méret/2 "piros
  kör_körvonal :méret :méret/10 "fehér
vége
```

### 2. variáció: Medál

A feladatot három, azonos vastagságú, de eltérő sugarú körvonal rajzolásával oldjuk meg. A körvonalak megrajzolása előtt el kell fordulnunk a teknőccel jobbra 90 fokot.

```
eljárás medál :méret
  jobbra 90
  kör_körvonal :méret 5 "kék
  kör_körvonal :méret*60/100 5 "zöld
  kör_körvonal :méret*30/100 5 "piros
  balra 90
vége
```

### 3. variáció: Virág

A virág alakzat 6 darab megrajzolt körvonalból áll. A körvonal megrajzolása után  $360/6=60$  fokot kell elfordulni a kívánt eredmény eléréséhez.

```
eljárás virág :méret
  ismétlés 6 [kör_körvonal :méret 5 "piros balra 60]
vége
```

## Kitöltött kör rajzolása (ha a teknőc pozíciója a körvonalon van)

Alapfeladat: Készíts olyan kitöltött körrajzoló eljárást, amely úgy rajzol kört, hogy a teknőc pozíciója a körvonalon helyezkedik el. Paraméterként lehessen megadni a kör sugarát, a körvonal színét és vastagságát, valamint a kitöltés színét.

```
eljárás töltöttkör_körvonal :sugár :vastagság :körvonalszín :kitöltésszín
  tollvastagság! :vastagság
  tollszín! :körvonalszín töltőszín! :kitöltésszín
  ismétlés 360 [előre :sugár*2*3,14159/360 jobbra 1]
  tollatfel jobbra 90 előre :sugár tölt hátra :sugár balra 90 tollatle
  töltőszín! 0 tollszín! 0
  tollvastagság! 1
vége
```

Próbáljuk ki az eljárást különböző paraméterekkel:



```
töltöttkör_körvonal 50 10 "piros
"sárga
```



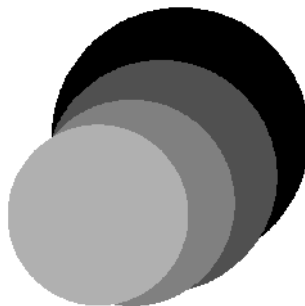
```
töltöttkör_körvonal 50 1 "piros
"piros
```

## Elemi feladatvariációk

A fenti eljárás segítségével oldd meg az alábbi feladatvariációkat:



1. variáció  
Színes körök



2. variáció  
Szürke korongok



3. variáció  
Turbán

### 1. variáció: Színek körök

Ebben a feladatban egyre csökkenő sugarú, megadott színű kitöltött köröket kell rajzolnunk.

```
eljárás színes_körök :méret
  töltöttkör_körvonal :méret 1 "fekete "fekete
  töltöttkör_körvonal :méret*80/100 1 "piros "piros
  töltöttkör_körvonal :méret*60/100 1 "sárga "sárga
  töltöttkör_körvonal :méret*40/100 1 "zöld "zöld
vége
```

## 2. variáció: Szürke korongok

Ha nem csak kirajzoljuk a különböző színű és eltérő sugarú köröket, hanem közben el is fordulunk a teknőccel, akkor megkaphatjuk a 2. variációként előállítható ábrát. Ebben az esetben állítsuk át a töltőmódot 1-es értékre, hogy az alakzat kitöltése az adott színű körvonalig történjen meg.

```
eljárás szürke_korongok :méret
  töltőmód! 1
  töltöttkör_körvonal :méret 1 "fekete "fekete jobbra 20
  töltöttkör_körvonal :méret*90/100 1 "szürke4 "szürke4 jobbra 20
  töltöttkör_körvonal :méret*80/100 1 "szürke6 "szürke6 jobbra 20
  töltöttkör_körvonal :méret*70/100 1 "szürke8 "szürke8 jobbra 20
  balra 80
  töltőmód! 0
vége
```

## 3. variáció: Turbán

Ha a kitöltött kör megrajzolása után elfordulunk adott szöggel, majd más színnel újból kirajzoljuk a kört, a lépéseket ismételve érdekes alakzatokat kapunk, mint például a feladatban szereplő turbánt.

```
eljárás turbán :méret
  jobbra 80 töltőmód! 1
  ismétlés 18 [töltöttkör_körvonal :méret 5 "piros "piros balra 10
              töltöttkör_körvonal :méret 5 "sárga "sárga balra 10]
  balra 80 töltőmód! 0
vége
```

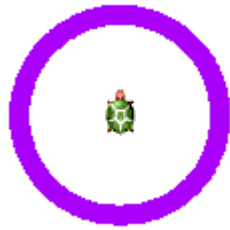
## **Kör rajzolása (ha a teknőc pozíciója a kör közepén van)**

A kör\_körvonal nevű eljárás végrehajtása után a teknőc a körvonalon helyezkedik el. Ez nem mindig szerencsés, számos probléma megoldása során előnyösebb lehet, ha a teknőc pozíciója a kör középpontját jelenti. Ezért gondoskodunk kell arról, hogy a teknőc felemelt tollal sugárnyi távolságot lépjen előre, és a körvonal megrajzolása után kerüljön vissza az eredeti pozíciójába.

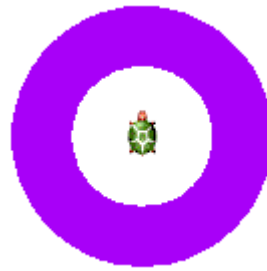
Alapfeladat: Készíts olyan eljárást, amely képes adott sugarú körvonal rajzolására, úgy hogy a teknőc kezdeti pozíciója a kör középpontját jelentse. Paraméterrel lehessen beállítani a körvonal színét és vastagságát!

```
eljárás kör_középpont :sugár :vastagság :körvonalszín
  tollvastagság! :vastagság tollszín! :körvonalszín
  tollatfel balra 90 előre :sugár jobbra 90 tollatle
  ismétlés 360 [előre :sugár*2*3,14159/360 jobbra 1]
  tollatfel balra 90 hátra :sugár jobbra 90 tollatle
  tollvastagság! 1 tollszín! 0
vége
```

Próbáld ki az eljárást különböző paraméterekkel:



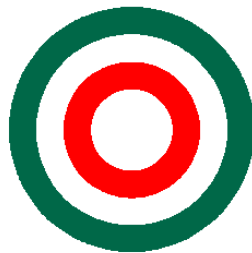
kör\_középpont 50 10 "lila



kör\_középpont 50 30 "lila

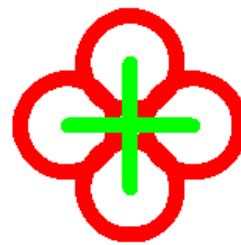
### Elemi feladatvariációk

A fenti eljárás segítségével oldjuk meg az alábbi feladatvariációkat:



1. variáció

Kokárda



2. variáció

Négy szirom

#### 1. variáció: Kokárda

A kokárda koncentrikus körökből áll, így a megrajzolt körök sugarát kell a feladatnak megfelelően beállítanunk. A körvonal vastagsága az átadott paraméter negyede lesz.

```
eljárás kokárda :méret
  kör_középpont :méret :méret/4 "sötétzöld3
  kör_középpont :méret*3/4 :méret/4 "fehér
  kör_középpont :méret/2 :méret/4 "piros
vége
```

#### 2. variáció: Négy szirom

Ebben a feladatban a négy kört úgy rajzoljuk meg, hogy a teknőccel (egy zöld vonalat rajzolva) előrelépünk, majd a kör megrajzolása után visszalépünk, és 90 fokos elfordulás után még háromszor megismételjük a rajzolást.

```
eljárás 4szirom :sugár
  ismétlés 4 [tollvastagság! :sugár/3 tollszín! "zöld előre :sugár*4/3
  kör_középpont :sugár :sugár/3 "piros
  tollszín! "zöld tollvastagság! :sugár/3 hátra :sugár*4/3
  jobbra 90]
vége
```



## Kitöltött kör rajzolása (ha a teknőc pozíciója a kör közepén van)

Alapfeladat: Készítsd el az eljárás azon változatát is, amely képes kitöltött kör rajzolására! Paraméterként lehessen megadni a kör sugarát, a körvonal színét és vastagságát, valamint a kitöltés színét!

```
eljárás töltöttkör_középpont :sugár :vastagság :körvonalszín :kitöltésszín
  tollvastagság! :vastagság
  tollszín! :körvonalszín töltőszín! :kitöltésszín
  tollatfel balra 90 előre :sugár jobbra 90 tollatle
  ismétlés 360 [előre :sugár*2*3,14159/360 jobbra 1]
  tollatfel balra 90 hátra :sugár tölt jobbra 90 tollatle
  tollvastagság! 1
  tollszín! 0 töltőszín! 0
vége
```

## Elemi feladatvariációk

A fenti eljárás segítségével oldjuk meg az alábbi feladatvariációkat:



1. variáció

Céltábla



2. variáció

Hold

### 1. variáció: Céltábla

A céltábla koncentrikus, kitöltött körökből áll, amelyek felváltva fekete illetve piros színnel vannak kitöltve.

```
eljárás céltábla :méret
  töltöttkör_középpont :méret 1 "fekete "fekete
  töltöttkör_középpont :méret*4/5 1 "piros "piros
  töltöttkör_középpont :méret*3/5 1 "fekete "fekete
  töltöttkör_középpont :méret*2/5 1 "piros "piros
  töltöttkör_középpont :méret/5 1 "fekete "fekete
vége
```

### 2. variáció: Hold

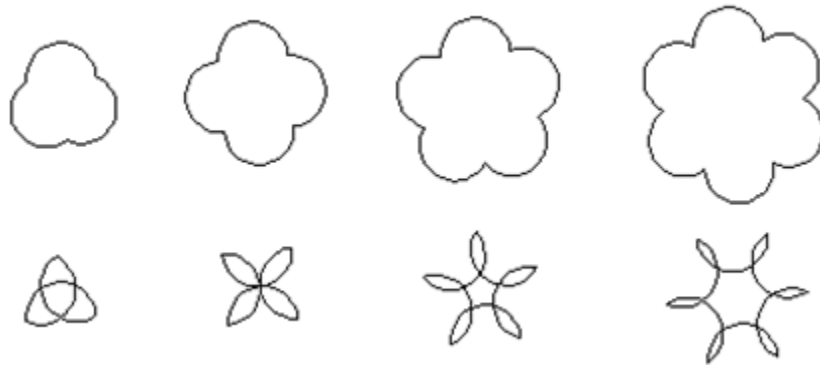
A hold alakot megrajzolhatjuk úgy, hogy egy fekete körvonalú, sárga kitöltésű körbe belerajzolunk egy fekete kitöltésű kört.

```
eljárás hold :méret
  töltöttkör_középpont :méret :méret/3 "fekete "sárga
  tollatfel jobbra 90 előre :méret/2 tollatle
  töltöttkör_középpont :méret/2 :méret/3 "fekete "fekete
  tollatfel hátra :méret/2 balra 90 tollatle
vége
```

## Félkör rajzolása

Vannak olyan esetek, amikor félkörök segítségével tudjuk előállítani a kívánt eredményt. Nézzük a következő alapfeladatot:

Alapfeladat: Készíts Logo eljárásokat (elsőábra :n :sugár, illetve másodikábra :n :sugár), amelyek megfelelő paraméterezéssel az alábbi ábrákat képesek rajzolni:



A félkör eljárást ebben az esetben megírhatjuk úgy, hogy a teknőc ne térjen vissza az eredeti pozíciójába, hiszen a végponttól kell folytatnunk a rajzolást. Az első ábra megrajzolásánál a félkör megrajzolása után jobbra fordulunk  $360/n$  szöggel.

```
eljárás félkör :sugár
  ismétlés 180 [előre :sugár*2*3,14159/180 jobbra 1] jobbra 180
vége
```

```
eljárás elsőábra :n :sugár
  ismétlés :n [félkör :sugár jobbra 360/:n]
vége
```

```
eljárás másodikábra :n :sugár
  ismétlés :n [félkör :sugár balra 360/:n]
vége
```

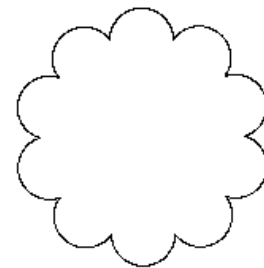
Az alábbi táblázatban láthatjuk, hogy milyen paraméterezéssel álltak elő az ábrák:



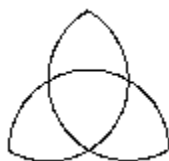
elsőábra 3 20



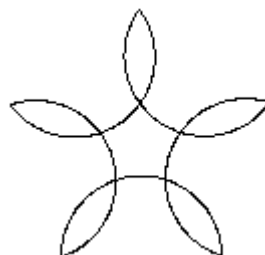
elsőábra 5 10



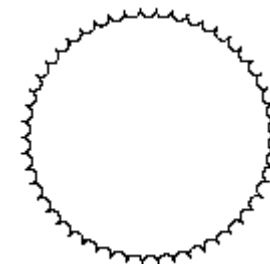
elsőábra 10 10



másodikábra 3 20



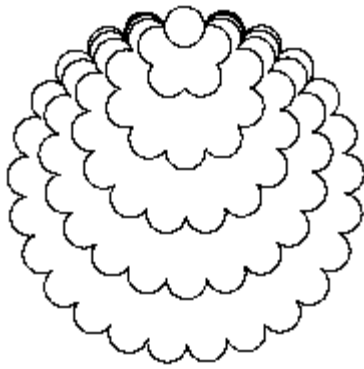
másodikábra 5 20



másodikábra 50 2

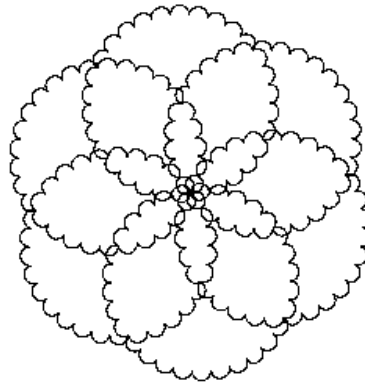
### Elemi feladatvariációk

A fenti eljárások segítségével oldjuk meg az alábbi feladatvariációkat:



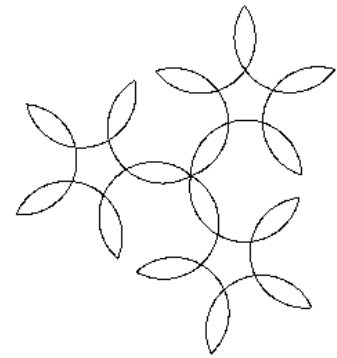
1. variáció

Fenyő



2. variáció

Asztalterítő



3. variáció

Levelek

#### 1. variáció

Ebben a variációban ugyanazon pozícióban kell egyre csökkenő méretű ábrákat rajzolni az elsőábra eljárás felhasználásával.

```
eljárás fenyő
  elsőábra 25 5
  elsőábra 20 5
  elsőábra 15 5
  elsőábra 10 5
  elsőábra 5 5
  elsőábra 2 5
vége
```

#### 2. variáció

Itt szintén az elsőábra eljárást használjuk fel, de a kirajzolás után el kell fordulnunk 60 fokkal.

```
eljárás asztalterítő
  ismétlés 6 [elsőábra 30 3 jobbra 60 ]
vége
```

#### 3. variáció

Ebben a variációban már a másodikábra eljárást használjuk ki úgy, hogy az ábra megrajzolása után elfordulunk 120 fokkal és még kétszer megismételjük a lépést.

```
eljárás levelek
  ismétlés 3 [másodikábra 5 20 jobbra 120]
vége
```

### Körívek rajzolása

Alapfeladat: Készítsd el a mintának megfelelően az olimpiai ötkarikát (ötkarika :r), ahol :r a színes 5 pont vastagságú körök sugara, a vízszintes körök :r/5 távolságra vannak egymástól, s az átfedéseknél 30 fokos szakadások vannak a körökön. A felső sor körei színe balról jobbra haladva: kék, fekete, piros; az alsó soré pedig: sárga, zöld.



A feladat megoldásához természetesen meg kell írunk először a körív rajzoló eljárást, amelynek paraméterül a kör sugarát (:r) és a körívhez tartozó középponti szöget, a körvonal vastagságát és színét kell megadunk.

```
eljárás körív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
```

Az eljárás eredménye különböző paraméterekkel:



```
körív 100 45 5 "piros      körív 100 120 10 "kék
```

```
eljárás ötkarika :r :vastag
  tollvastagság! :vastag balra 90
  kékkör :r :vastag tollatfel hátra 2*:r+:r/5
  tollatle feketekör :r :vastag tollatfel hátra 2*:r+:r/5
  tollatle piroskör :r :vastag jobbra 90
  tollatfel hátra :r+:r/5 balra 90 előre :r
  tollatle zöldkör :r :vastag tollatfel előre 2*:r+:r/5
  tollatle sárgakör :r :vastag
  tollvastagság! 1 tollszín! 0
  tollatfel előre :r+:r/5 jobbra 90 előre :r+:r/5 tollatle
vége
```

```
eljárás feketekör :r :vastag
  körív :r 60 :vastag "fekete tollatfel
  körív :r 30 :vastag "fekete tollatle
  körív :r 240 :vastag "fekete tollatfel
  körív :r 30 :vastag "fekete tollatle
vége
```

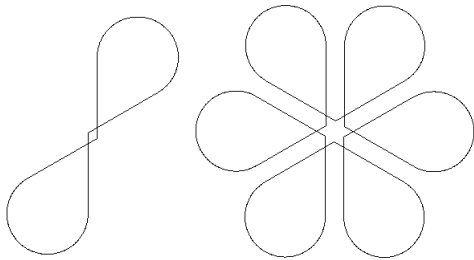
```
eljárás kékkör :r :vastag
  körív :r 330 :vastag "kék tollatfel
  körív :r 30 :vastag "kék tollatle
vége
```

```
eljárás piroskör :r :vastag
  körív :r 60 :vastag "piros tollatfel
  körív :r 30 :vastag "piros tollatfel tollatle
  körív :r 270 :vastag "piros tollatfel
vége
```

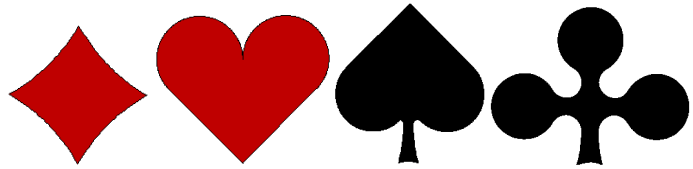
```
eljárás sárgakör :r :vastag
körív :r 150 :vastag "sárga tollatfel
körív :r 30 :vastag "sárga tollatle
körív :r 60 :vastag "sárga tollatfel
körív :r 30 :vastag "sárga tollatle
körív :r 90 :vastag "sárga
vége
```

```
eljárás zöldkör :r :vastag
körív :r 150 :vastag "zöld tollatfel
körív :r 30 :vastag "zöld tollatle
körív :r 60 :vastag "zöld tollatfel
körív :r 30 :vastag "zöld tollatle
körív :r 90 :vastag "zöld
vége
```

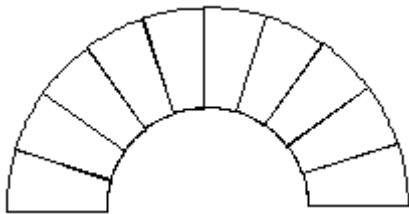
### Feladatvariációk



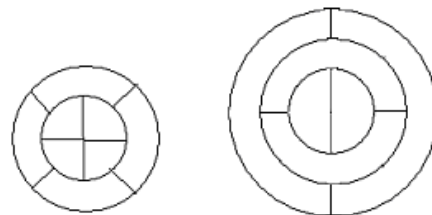
1. variáció  
Díszek



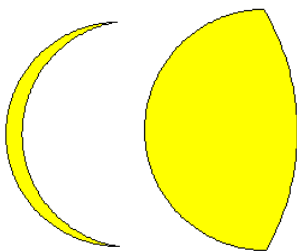
2. variáció  
Kártyák



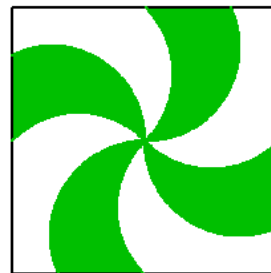
3. variáció  
Kapu



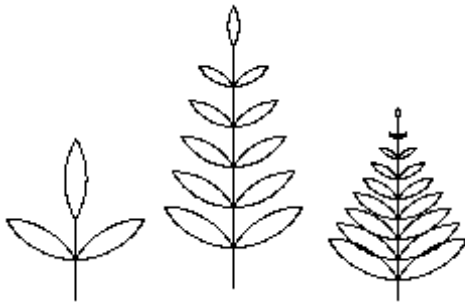
4. variáció  
Kőlap



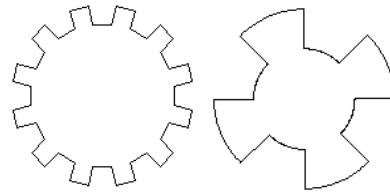
5. variáció  
Holdak



6. variáció  
Festett ívek egy négyzetben



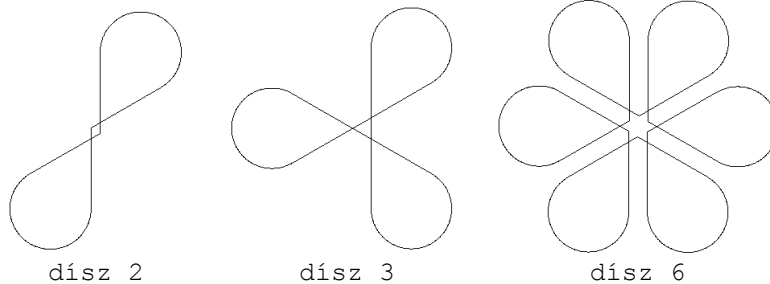
7. Variáció  
Levelek



8. Variáció  
Fogaskerek

1. variáció: Díszek

Gótikus ajtódíszekben bizonyos mintákat ismételhetnek, egymáshoz képest elforgatva. Készíts Logo eljárást (dísz), amely az alábbi típusú ábrákat rajzolja!



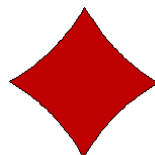
```

eljárás dísz :n
  ismétlés :n [alap 50 jobbra 120-360/:n]
vége

eljárás alap :d
  előre 2*:d körív :d 240 1 "fekete előre 2*:d
vége
    
```

2. variáció: Franciakártya szimbólumok rajzolása

Készítsd el az alábbi franciakártya szimbólumokat! A kőr lapban az egyenes vonalak hossza a körívek sugarának  $(1+\sqrt{2})$ -szöröse legyen! A pikk lapban az egyenes vonalak hossza az oldalsó körívek sugarának  $(1+\sqrt{2})$ -szöröse legyen! A pikk-ben a többi körív sugara vagy négyszerese, vagy nyolcadrésze a két oldalsó körívének, vagy pedig egyenlő vele. A treffben a három majdnem teljes kör sugarának fele vagy pedig négyszerese a többi körív sugara.



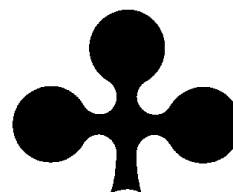
káró



kőr



pikk



treff

A feladat megvalósítása egyszerűbbé válik, ha elkészítjük a körív rajzoló eljárásunk azon változatát is, amely balra fordulva rajzolja meg a körívet. Hívjuk ez bkörív-nek!

```
eljárás bkörív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 balra 1]
vége
```

```
eljárás káró :h
  balra 30 ismétlés 4 [bkörív 2*:h 30 1 "fekete jobbra 120]
  tollatfel előre :h/2 töltőszín! "piros tölt hátra :h/2 tollatle
  jobbra 30
vége
```

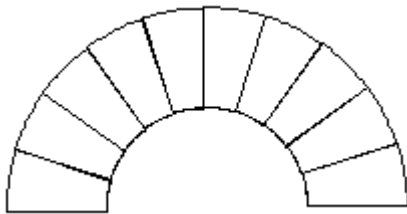
```
eljárás kör :r
  balra 45 előre :r*(1+gyök 2)
  körív :r 225 1 "fekete balra 180 körív :r 225 1 "fekete
  előre :r*(1+gyök 2) jobbra 90 jobbra 45
  tollatfel előre :r töltőszín! "piros tölt hátra :r tollatle
vége
```

```
eljárás pikk :r
  balra 180 körív :r 225 1 "fekete előre :r*(1+gyök 2) jobbra 90
  előre :r*(1+gyök 2) körív :r 180 1 "fekete bkörív :r/8 135 1 "fekete
  bkörív 4*:r 15 1 "fekete
  jobbra 120 bkörív :r 30 1 "fekete jobbra 120 bkörív 4*:r 15 1 "fekete
  bkörív :r/8 135 1 "fekete jobbra 135 tollatfel előre :r
  töltőszín! "fekete tölt
  hátra :r tollatle
vége
```

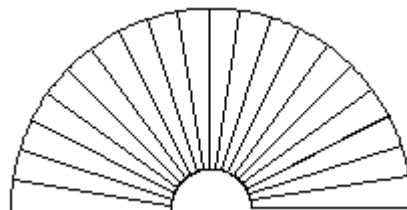
```
eljárás treff :r
  balra 90 bkörív :r 22,5 1 "fekete jobbra 127,5 bkörív 4*:r 15 1 "fekete
  bkörív :r/2 150 1 "fekete körív :r 300 1 "fekete
  bkörív :r/2 210 1 "fekete körív :r 300 1 "fekete
  bkörív :r/2 210 1 "fekete körív :r 300 1 "fekete
  bkörív :r/2 150 1 "fekete bkörív 4*:r 15 1 "fekete jobbra 127,5
  bkörív :r 22,5 1 "fekete jobbra 90
  tollatfel előre :r töltőszín! "fekete tölt hátra :r tollatle
vége
```

### 3. variáció

Készíts Logo programot (kapu :rk :rb :db), amely egy körív alakú kaput rajzol! A külső körív :rk, a belső pedig :rb sugarú legyen! A kapu :db darab részből álljon!



kapu 100 50 10



kapu 100 20 20

```
eljárás kapu :rk :rb :db
  ismétlés :db [balra 90 előre :rk-:rb jobbra 90
    körív :rk 180/:db 1 "fekete jobbra 90
    előre :rk-:rb jobbra 90 bkörív :rb 180/:db 1 "fekete
    jobbra 180 körív :rb 180/:db 1 "fekete]
  tollatfel balra 90 hátra 2*:rb balra 90 tollatle
vége
```

```

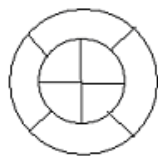
eljárás bkörív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 balra 1]
vége
  
```

```

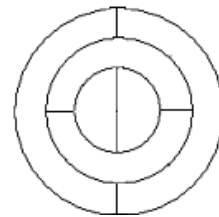
eljárás körív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
  
```

#### 4. variáció

Kör alakú területet szeretnénk lefedni mozaik lapokkal. Mindegyik körgyűrűben pontosan :rész darab lapot helyezünk el, a tőle beljebb levő körgyűrűhöz képest fél résszel elforgatva. Írj Logo eljárást (kőlapok :r :rész :db), amely :r sugarú körből indulva még :db-1 darab :r szélességű körgyűrűt rajzol, mindegyiket :rész darabszámú részre osztva!



kőlap 20 4 2



kőlap 20 2 3

Kívülről befelé rajzoljuk a köröket, körívekből összeállítva azokat. A körök kirajzolása előtt mindig rajzolunk egy fele akkora belső szöggel rendelkező ívet is, így az egyes sávok egymástól elforgatva látszanak.

```

eljárás kőlap :r :rész :db
  ha :db>0 [körív :r*:db 360/:rész/2 "fekete 1
    ismétlés :rész [körív :r*:db 360/:rész "fekete 1 jobbra 90
      előre :r hátra :r balra 90]
    jobbra 90 előre :r balra 90 kőlap :r :rész :db-1]
vége
  
```

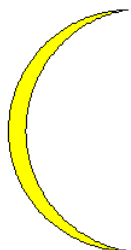
```

eljárás körív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
  
```

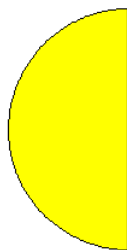
#### 5. variáció: Holdak rajzolása

Gyerekek úgy rajzolnak holdat, hogy egy félkör belsejébe egy nagyobb sugarú körívet rajzolnak. Ha a félkör sugara  $R$  és a belső körív  $A$  fokkal jobbra kezdődik a félkörhöz képest, akkor a belső körív sugara:  $R/\cos(A)$ , az ívhez tartozó középponti szög pedig  $180-2*A$ .

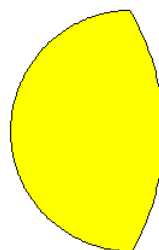
Alapfeladat: Készíts holdat rajzoló eljárást (hold :x :y), ahol :x a hold külső íve sugara, :y pedig a belső ív dőlésszöge a körhöz képest!



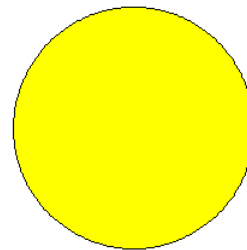
hold 100 10



hold 100 90



hold 100 120



hold 100 180



A feladat megoldásához természetesen a korábban megírt körív rajzoló eljárást használjuk fel. A rajzoláskor érdemes elkülöníteni a 90 fok alatti, a pontosan 90 fokos, és 90 fok feletti középponti szögű eseteket, így a fok alapján érdemes elágazást készíteni.

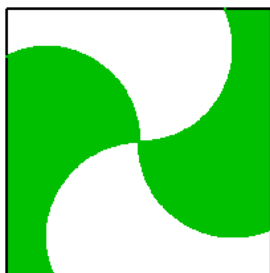
```

eljárás hold :r :fok
  balra 90 körív :r 180 1 "fekete
  tollatfel jobbra 90 előre 2*:r jobbra 90 tollatle
  ha :fok<90 [jobbra :fok körív :r/cos :fok 180-2*:fok 1 "fekete tollatfel
              jobbra 90+:fok előre 2*:r jobbra 90 tollatle]
  ha :fok=90 [jobbra 90 előre 2*:r hátra 2*:r balra 90]
  ha :fok>90 [tollatfel jobbra 90 előre 2*:r jobbra 270-:fok
              tollatle körív :r/cos (180-:fok) 2*:fok-180 1 "fekete
              jobbra 180-:fok]
  tollatfel körív :r 90 1 "fekete jobbra 90 előre 5 töltőszín! "sárga tölt
  hátra 5 balra 90 körív :r 270 1 "fekete jobbra 90 tollatle
vége

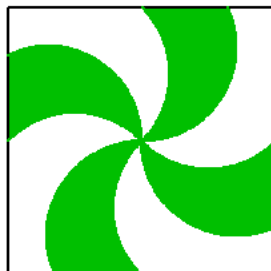
```

### 6. variáció: Festett ívek egy négyzetben

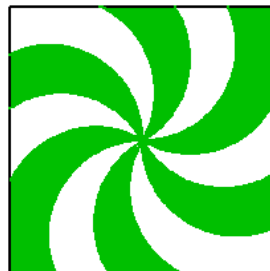
Készíts Logo eljárást (ábra :hossz :db), amely egy  $2 \cdot \text{hossz}$  méretű négyzetbe :db darab körívvel határolt zöld területet fest! A körívek egy  $\text{hossz} \cdot \sqrt{2}$  átmérőjű kör megfelelő körívei.



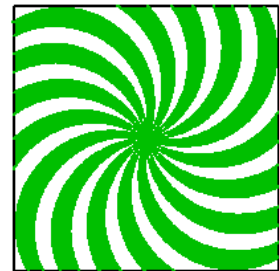
ábra 100 2



ábra 100 4



ábra 100 6



ábra 100 16

A feladat megoldása során meg kell oldani, hogy csak a négyzetben belül legyen az ábra megrajzolva. A belül? függvény visszaadja, hogy a teknőc alakja (részben vagy egészben) benne van a paraméterül átadott téglalap területében. Mivel a teknőc alakja befolyásolja ezen függvény által visszaadottakat, érdemes a teknőc alakját minél kisebbre állítani. Ezt meg tudjuk tenni az alak! [pontméret 1] paranccsal. Ekkor a teknőc alakja egy pont lesz.

```

eljárás ábra :h :n
  látható! "hamis alak! [pontméret 1] tollszín! "fekete
  tollatfel hátra :h jobbra 90 előre :h balra 90
  tollatle négyzet 2*:h
  tollatfel jobbra 90 hátra :h balra 90 előre :h tollatfel
  ismétlés :n [ívek :h :n hova :h jobbra 360/:n]
  tollszín! "fekete látható! "igaz tollatfel
  hátra :h jobbra 90 előre :h balra 90 tollatle négyzet 2*:h
  alak! []
vége

eljárás négyzet :h
  ismétlés 4 [előre :h balra 90]
vége

```

Az alábbi eljárás az ívek között kifesti a területet.

```

eljárás ívek :h :n :t
  tollszín! "zöld töltőszín! "zöld
  ívonalig :h :t balra 180/:n ívonalig :h :t jobbra 90/:n
  tollatfel ismétlés 20 [előre :h*(gyök 2)*3,14159/360 balra 1]
  tölt
  ismétlés 20 [jobbra 1 hátra :h*(gyök 2)*3,14159/360]
  jobbra 90/:n
vége
  
```

A következő eljárás rajzolja meg a négyzet közepétől az oldalig húzott ívet, amely először felemelt tollal „tapogatja ki” az utat, majd visszahátrál a kiinduló pontig.

```

eljárás ívonalig :h :ter
  tollatfel előre :h*(gyök 2)*3,14159/3600 balra 0,1
  ha belül? :ter [ívonalig :h :ter]
  tollatle jobbra 0,1 hátra :h*(gyök 2)*3,14159/3600
vége
  
```

```

eljárás hova :h
  eredmény (lista xpoz-:h ypoz+:h 2*:h 2*:h)
vége
  
```

### 7. variáció: Levelek

Egy összetett levél sok kisebb levélből áll. A száron az egyes levelek 5-tel kisebb méretűek az őket megelőző levélpárnál (a körív 5-tel kisebb sugarú körből készül). Készíts Logo programot (levelek :n :h :t), amely :n levélpárt rajzol, egymástól :t távolságra! Az első levélpár körívének sugara legyen :h, a legvégén pedig egyetlen levél álljon!



levelek 1 50 20

levelek 5 50 20

levelek 8 50 10

```

eljárás levelek :n :h :t
  előre :t balra 30 blevél :h jobbra 60 jlevél :h balra 30
  ha :n>1 [levelek :n-1 :h-5 :t]
  ha :n=1 [előre :t jobbra 30 blevél :h balra 30 hátra :t]
  hátra :t
vége
  
```

Praktikus a levél eljárást kétféleképpen megírni, az egyikben balívekből, a másikban jobbívekből építjük fel.

```

eljárás blevél :r
  bkörív :r 60 "fekete 1 balra 120 bkörív :r 60 "fekete 1 balra 120
vége
  
```

```

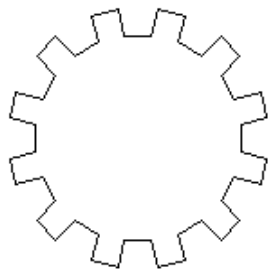
eljárás jlevél :r
  körív :r 60 "fekete 1 jobbra 120 körív :r 60 "fekete 1 jobbra 120
vége
  
```

```
eljárás bkörív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 balra 1]
vége
```

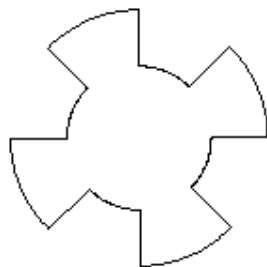
```
eljárás körív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
```

### 8. variáció

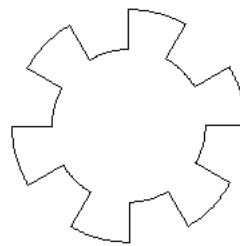
Egy gyárban kétféle típusú fogaskereket gyártanak. Az A típusnál a fogakat és a keréktárcsát egyenes szakaszok határolják, a B típusnál pedig a tárcsa és a fogak külső felülete is körív alakú. Készíts Logo eljárásokat (Foga, Fogb), melyek a kétféle típusú fogaskereket rajzolják, ha paraméterül a rajzolendő fogak számát és hosszát adjuk nekik!



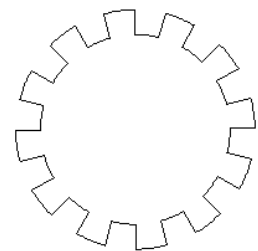
Foga 12 20



Fogb 4 50



Fogb 6 50



Fogb 12 50

```
eljárás Foga :n :h
  ismétlés :n [előre :h balra 90-180/:n
              ismétlés 2 [előre :h jobbra 90]
              előre :h balra 90-180/:n]
vége
```

```
eljárás Fogb :n :h
  ismétlés :n [körív :h*:n/3,14159/2 180/:n "fekete 1 balra 90 előre :h/2
              jobbra 90 körív (:h*:n/3,14159+:h)/2 180/:n "fekete 1
              jobbra 90 előre :h/2 balra 90]
vége
```

```
eljárás körív :r :fok :vastag :szín
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
```

## Variációk zászlók rajzolására

A különböző országok lobogói felépítésükben, színvilágukban, de akár oldalarányaikban is rendkívül különbözőek lehetnek. Erről magunk is meggyőződhetünk, ha megtekintjük a <http://bit.ly/1Hae1BH> weboldalt.

Ebben a leckében különböző országok lobogóit fogjuk elkészíteni Logo eljárások segítségével úgy, hogy a lobogók tetszőleges méretben megjeleníthetők legyenek.

Megjegyzés: Ahhoz, hogy a zászlókon található elemek méretét le tudjuk mérni, és később a zászló méretéhez viszonyítva százalékosan meg tudjuk adni, érdemes a zászlót a vágólappra tenni és beilleszteni egy olyan képszerkesztő programba, amelyben könnyen leolvashatóak a méretek, illetve koordináták. Ilyen például a Logomotion, vagy GIMP szerkesztőprogram. A képet érdemes 100 képpont szélességűre, illetve magasságúra átméretezni, így a leolvasott méretek egyből felhasználhatóak a százalékos megadás esetén. Ugyanezen alkalmazásokban a zászlón szereplő színek színkódját is meg tudjuk határozni.

Alapfeladat: Készítsünk olyan logó eljárást, amely képes kirajzolni Bolívia zászlaját, amely azonos magasságú, színezett téglalapokból áll és fekete szegéllyel rendelkezik! A lobogó magasságának és szélességének aránya: 15:22, vagyis a szélesség a magasság 1,467-szorososa.



Bolívia

A feladat megoldásához érdemes olyan téglalap rajzoló eljárást készíteni, amely adott magasságú, szélességű és színű téglalapot rajzol (téglalap eljárás).

```
eljárás téglalap :magas :széles :szín
  tollszín! :szín tolllatle
  ismétlés 2 [előre :magas jobbra 90 előre :széles jobbra 90]
  jobbra 45 tollatfel előre 5 töltőszín! :szín tölt hátra 5
  tollatle balra 45
vége
```

Mivel a lobogónak szegélye is van, olyan téglalap rajzolására is szükség van, amely adott színű és vastagságú körvonallal rendelkezik, és nincs kitöltve (szegély eljárás).

```
eljárás szegély :magas :széles :szín :vastagság
  tollszín! :szín tolllatle tollvastagság! :vastagság
  ismétlés 2 [előre :magas jobbra 90 előre :széles jobbra 90]
  tollvastagság! 1
vége
```

E kettő eljárást természetesen egy eljárásnéven is megírhatnánk, megadva a körvonal színét és a kitöltés színét, azonban a jobb átláthatóság érdekében példáinkban e két külön eljárást fogjuk használni.

A Bolívia eljárás paramétereként adjuk meg a zászló szélességét. A magasságot nem feltétlenül kell megadni, mert az kiszámítható a szélességből a lobogó oldalarányainak ismeretében.

A zászló mérete alapján fogjuk kiszámítani, hogy milyen magas és széles csíkokat kell kirajzolnunk. Jelen esetben a csíkok magassága a zászló magasságának harmada lesz. A szegély megrajzolását érdemes a legvégén elvégezni, a takarás miatt.

Egy lobogónál nagyon fontos, hogy az azokon szereplő színárnyalatokat használjuk. Az Imagine Logo-ban van lehetőség RGB színkoordináta rendszer szerint megadni a színeket. Más logóverziókban ez a lehetőség nem feltétlenül adott, ott kénytelenek vagyunk a rendelkezésre álló színek közül kiválasztani a legoptimálisabbat.

Bolívia lobogóját tekintve a színek kódjai: piros [213 43 30], sárga [249 227 0], zöld [0 121 52].

```
eljárás bolíviazászló :széles
  bolívia :széles :széles*15/22
vége
```

Megjegyzés: a későbbiekben feltesszük, hogy az egyes zászló rajzoló eljárásokat a szabályos szélesség:magasság arányú paraméterekkel hívjuk meg.

```
eljárás bolívia :széles :magas
  téglalap :magas/3 :széles [0 121 52]
  tollatfel előre :magas/3 tollatle
  téglalap :magas/3 :széles [249 227 0]
  tollatfel előre :magas/3 tollatle
  téglalap :magas/3 :széles [213 43 30]
  tollatfel hátra 2*:magas/3 tollatle
  szegély :magas :széles "fekete 4
vége
```

A következőkben különböző feladatsorok segítségével egyre változatosabb, komplexebb lobogókat állítunk elő.

- Vízszintes csíkok alkalmazásával állítunk elő zászlókat
- Kombináljuk a vízszintes csíkokat a függőleges csíkokkal
- A függőleges és vízszintes csíkok alkalmazásának speciális esetét is áttekintjük, amikor keresztet kell rajzolnunk.
- Készítünk olyan lobogókat, amelyeken a kör, mint síkidom kap fő szerepet.
- Olyan zászlókat készítünk, amelyeken – a korábbi elemek mellett - csillag motívum látható.
- Végül pedig háromszögeket tartalmazó lobogókat valósítunk meg.

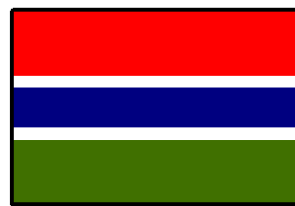
## 1. feladatsor: az alapmegoldás programjából kiindulva

A módszer lényege, hogy az alapfeladat megoldását módosítsuk úgy, hogy újabb feladatokat állítsunk elő belőle. Ezek nehézségi szintje változhat attól, hogy hány csíkból álló lobogót készítünk.

### Elemi feladatvariációk:



Mauritius



Gambia



Botswana



Thaiföld

#### 1. variáció: Mauritius zászlaja

Mauritius zászlajának felépítése nagyon hasonló az alapfeladatban látottéhoz, csak három helyett négy egyenlő magasságú téglalapot kell rajzolnunk. Az oldalarány ebben az esetben (illetve a többi variációt tekintve is) 2:3, vagyis a szélesség másfélszer akkora, mint a magasság. A színek kódjai: piros [234 40 57], kék [26 32 109], sárga [255 213 0], zöld [0 165 81].



```
eljárás mauritius :széles :magas
  téglalap :magas/4 :széles [0 165 81]
  tollatfel előre :magas/4 tollatle
  téglalap :magas/4 :széles [255 213 0]
  tollatfel előre :magas/4 tollatle
  téglalap :magas/4 :széles [26 32 109]
  tollatfel előre :magas/4 tollatle
  téglalap :magas/4 :széles [234 40 57]
  tollatfel hátra 3*:magas/4 tollatle
  szegély :magas :széles "fekete 4
vége
```

#### 2/a variáció: Gambia zászlaja

Gambia zászlaja már nem azonos magasságú téglalapokból áll. Az ismétlődő csíkokat azonban érdemes egymásba rajzolt téglalapokkal megvalósítani. Vagyis ahelyett, hogy az alsó zöld téglalap megrajzolása után rajzolnánk egy fehér téglalapot, majd egy kék téglalapot, és újra egy fehér téglalapot, inkább rajzoljunk egy fehér kitöltésű téglalapot, amelynek belsejében rajzoljuk meg a kék téglalapot. Így öt téglalap rajzolás helyett négy téglalap rajzolással oldjuk meg a feladatot. A zöld, fehér és piros csíkok magassága a zászló magasságának harmada. Az oldalarány: 2:3.



A színek kódjai: piros [206 17 38], kék [12 28 140], zöld [58 119 40].

```
eljárás gambia :széles :magas
téglalap :magas/3 :széles [58 119 40]
tollatfel előre :magas/3 tollatle
téglalap :magas/3 :széles "fehér
tollatfel előre :magas/3/5 tollatle
téglalap :magas/3*3/5 :széles [12 28 140]
tollatfel előre :magas/3*4/5 tollatle
téglalap :magas/3 :széles [206 17 38]
tollatfel hátra 2*:magas/3 tollatle
szegély :magas :széles "fekete 4
vége
```

### 2/b variáció: Botswana zászlaja

Az előbbi variáció arányainak és színeinek módosításával eljutunk Botswana lobogójához. Itt a két kék csík magassága a zászló magasságának 37%-a lesz, a középső csík pedig 26% magasságú. A középső fehér csíkot 20%-60%-20% arányban osztjuk tovább fehér-fekete-fehér csíkokra. Az oldalarány itt is 2:3. A kék szín kódja: [117 170 219]



```
eljárás botswana :széles :magas
téglalap :magas*40/100 :széles [117 170 219]
tollatfel előre :magas*40/100 tollatle
téglalap :magas*20/100 :széles "fehér
tollatfel előre :magas*20/100*20/100 tollatle
téglalap :magas*20/100*60/100 :széles "fekete
tollatfel előre :magas*20/100*80/100 tollatle
téglalap :magas*40/100 :széles [117 170 219]
tollatfel hátra :magas*60/100 tollatle
szegély :magas :széles "fekete 4
vége
```

### 3. variáció: Thaiföld lobogója

Thaiföld zászlajának megrajzolása is hasonló módon történik, de ebben az esetben nem rajzolunk szegélyt a lobogó köré. A középső fehér csík a zászló magasságának kétharmada legyen, az alsó és felső piros csík pedig a maradék egyharmad rész fele.



A rajzoláskor megtehetjük, hogy kezdetben a teljes területet pirossal töltsük ki, majd megrajzoljuk a fehér sávot, amelynek közepére helyezük a kék területet. A kék csík a fehér csík magasságának 50%-a legyen.

Az oldalarány: 2:3. A színek kódjai: piros [237 28 36], kék [36 29 79].

```
eljárás thaifold :széles :magas
téglalap :magas :széles [237 28 36]
tollatfel előre :magas/3/2 tollatle
téglalap :magas*2/3 :széles "fehér
tollatfel előre :magas*2/3*25/100 tollatle
téglalap :magas*2/3*50/100 :széles [36 29 79]
tollatfel hátra :magas/3/2 hátra :magas*2/3*25/100 tollatle
vége
```

Milyen más lobogók vannak még, amelyeket vízszintes csíkokkal lehetne megrajzolni? (Segítség: [http://en.wikipedia.org/wiki/List\\_of\\_countries\\_by\\_proportions\\_of\\_national\\_flags](http://en.wikipedia.org/wiki/List_of_countries_by_proportions_of_national_flags))

## 2. feladatsor: Vízszintes és függőleges csíkok együttes alkalmazása

A következőkben nézzünk komplexebb lobogókat, ahol nemcsak vízszintes sávokat használunk, hanem függőlegeseket is. Kezdjük Belgium zászlajával, amely három azonos szélességű téglalapról áll! A lobogó oldalaránya: 13:15.

Alapfeladat: Belgium1

A lobogót megrajzolhatjuk úgy, hogy a teknőc végig a zászló bal alsó sarkában marad, és kezdetben egy teljes méretű piros téglalapot rajzol, majd egy keskenyebb sárgát, és egy még keskenyebb feketét, felülrajzolva a korábbi téglalapokat.



A színek kódok: sárga [250 224 66], piros [237 41 57].

```
eljárás belgium1 :széles :magas
  téglalap :magas :széles [237 41 57]
  téglalap :magas :széles*2/3 [250 224 66]
  téglalap :magas :széles/3 "fekete
vége
```

Alapfeladat: Belgium2

A másik módszer, hogy mindig ugyanakkora téglalapokat rajzolunk és a teknőcünket léptetjük jobbra, majd a végén gondoskodunk arról, hogy a teknőc felvegye a kiindulási állapotát és irányát.

```
eljárás belgium2 :széles :magas
  téglalap :magas :széles/3 "fekete
  tollatfel jobbra 90 előre :széles/3 balra 90 tollatle
  téglalap :magas :széles/3 [250 224 66]
  tollatfel jobbra 90 előre :széles/3 balra 90 tollatle
  téglalap :magas :széles/3 [237 41 57]
  tollatfel jobbra 90 hátra :széles*2/3 balra 90 tollatle
vége
```

### Elemi feladatvariációk:



Benin



Dánia



Izland

1/a variáció: Benin1

Benin zászlaja esetén is kipróbálhatjuk az előző módszereket. Az oldal-arány ebben az esetben: 2:3. A zöld téglalap szélessége a lobogó szélességének 40%-a. Nézzük először azt a megoldást, amikor a teknőc nem mozdul el a bal alsó sarokból!



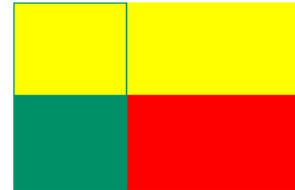
A színek kódok: piros [232 17 45], sárga [252 209 22], zöld [0 135 81].



```
eljárás beninlrossz :széles :magas
  téglalap :magas :széles [252 209 22]
  téglalap :magas/2 :széles [232 17 45]
  téglalap :magas :széles*40/100 [0 135 81]
vége
```

A program futtatásakor észrevehetjük, hogy nem a kívánt eredményt kaptuk.

A bal oldali téglalap csak részben lett zöld színnel kitöltve. Ennek magyarázata az, hogy a tölt eljárás alapesetben csak az azonos színű területeket képes kitölteni a teknőc tollszínével megrajzolt határon belül. Mivel ebben az esetben a zölddel megrajzolt téglalap belsejében piros és sárga terület is van, csak az egyik kerül kitöltésre. Emiatt külön gondoskodnunk kellene arról, hogy a teknőccel a kitöltetlen területre lépünk, majd újra kiadjuk a tölt parancsot. Azonban van más megoldás is. Módosíthatjuk a terület kitöltési szabályt úgy, hogy a teknőc tollszínével megrajzolt határig történjen a kitöltés. Ehhez a töltőmód! 1 parancsot kell kiadni. Az alapértelmezett kitöltési mód visszaállításához pedig a töltőmód! 0 parancs használható.



Ez alapján a módosított eljárás:

```
eljárás benin1 :széles :magas
  töltőmód! 1
  téglalap :magas :széles [252 209 22]
  téglalap :magas/2 :széles [232 17 45]
  téglalap :magas :széles*40/100 [0 135 81]
  töltőmód! 0
vége
```

**Kérdés:** Belgium zászlójának megrajzolásakor helyes eredményt kaptunk. Abban az esetben miért nem kellett a töltőmód beállítását módosítani?

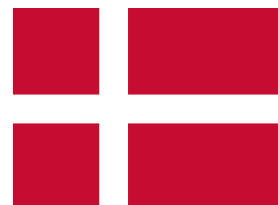
### 1/b variáció: Benin2

A lobogót persze úgy is megrajzolhatjuk, hogy adott szélességű zöld téglalapot rajzolunk, majd mellé rajzoljuk a további két téglalapot is.

```
eljárás benin2 :széles :magas
  téglalap :magas :széles*40/100 [0 135 81]
  tollatfel jobbra 90 előre :széles*40/100 balra 90
  téglalap :magas :széles*60/100 [252 209 22]
  téglalap :magas/2 :széles*60/100 [232 17 45]
  tollatfel jobbra 90 hátra :széles*40/100 balra 90
vége
```

### 2. variáció: Dánia

Dánia lobogójának megrajzolását többféle módon is elvégezhetjük. Felfoghatjuk úgy, hogy 4 különböző méretű, piros kitöltésű téglalapot kell rajzolnunk, vagy úgy is, hogy egy piros kitöltésű téglalapba kell egy vízszintes és (a kitöltési probléma miatt) két függőleges csíkot rajzolnunk. Nézzük meg mindkettőt! Dánia zászlójának oldalaránya: 28:37. A piros szín kódja: [198 12 48].



A bal oldali téglalapok a zászló szélességének 32%-ánál érnek véget, a fehér csík 11%-nyi széles, így a jobb oldali téglalap a szélesség 43%-ánál kezdődik. Függetlenül a méretek a következőképpen alakulnak. A felső téglalapok magassága a lobogó magasságának 43%-a, a csík magassága a magasság 14%-a, az alsó téglalapok magassága 43%.

### 2/a variáció: Dánia – 4 téglalap

```
eljárás dánia_4téglala :széles :magas
  téglalap :magas :széles "fehér
  téglalap :magas*43/100 :széles*32/100 [198 12 48]
  tollatfel előre :magas*57/100 tollatle
  téglalap :magas*43/100 :széles*32/100 [198 12 48]
  tollatfel jobbra 90 előre :széles*43/100 balra 90 tollatle
  téglalap :magas*43/100 :széles*57/100 [198 12 48]
  tollatfel hátra :magas*57/100 tollatle
  téglalap :magas*43/100 :széles*57/100 [198 12 48]
  tollatfel jobbra 90 hátra :magas*57/100 balra 90 tollatle
vége
```

### 2/b variáció: Dánia – fehér csíkok

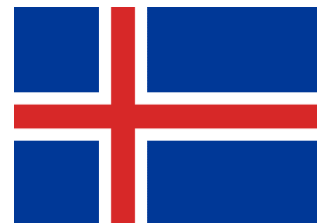
```
eljárás dánia :széles :magas
  téglalap :magas :széles [198 12 48]
  tollatfel előre :magas*43/100 tollatle
  téglalap :magas*14/100 :széles "fehér
  tollatfel hátra :magas*43/100
  jobbra 90 előre :széles*32/100 balra 90 tollatle
  téglalap :magas*43/100 :széles*11/100 "fehér
  tollatfel előre :magas*57/100 tollatle
  téglalap :magas*43/100 :széles*11/100 "fehér
  tollatfel hátra :magas*57/100 jobbra 90 hátra :széles*32/100 balra 90
  tollatle
vége
```

### 3. variáció: Izland

Izland lobogójának oldalaránya: 18:25. A színek kódok: kék [0 56 151], piros [215 40 40].

A bal oldali kék téglalapok a szélesség 28%-ának felelnek meg. A fehér sáv 16% széles, a jobb oldali téglalapok szélessége így 56%. A piros csík szélessége a lobogó szélességének 8%-a.

A felső téglalapok magassága a lobogó magasságának 39%-a. A fehér csík magassága 22%, a piros csík 11%, az alsó téglalapé 39%.



Ebben az esetben a függőleges piros csíkot egyben rajzoljuk meg, amelynek alsó részét rendben ki is tölti a téglalap eljárásunk, a felső rész kitöltését pedig a teknőc pozícionálásával és a tölt parancs használatával oldjuk meg.

```
eljárás izland :széles :magas
  kéktéglalap :magas :széles
  fehérkereszt :magas :széles
  piroskereszt :magas :széles
vége
```

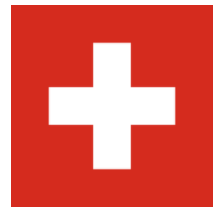
```
eljárás kéktéglalap :magas :széles
  téglalap :magas :széles [0 56 151]
vége
```

```
eljárás fehérkereszt :magas :széles
tollatfel előre :magas*39/100 tollatle
téglalap :magas*22/100 :széles "fehér
tollatfel hátra :magas*39/100
jobbra 90 előre :széles*28/100 balra 90 tollatle
téglalap :magas*39/100 :széles*16/100 "fehér
tollatfel előre :magas*61/100 tollatle
téglalap :magas*39/100 :széles*16/100 "fehér
tollatfel hátra :magas*61/100 jobbra 90 hátra :széles*28/100 balra 90
vége
```

```
eljárás piroskereszt :magas :széles
előre :magas*44,5/100 tollatle
téglalap :magas*11/100 :széles [215 40 40]
tollatfel hátra :magas*44,5/100
jobbra 90 előre :széles*32/100 balra 90 tollatle
téglalap :magas :széles*8/100 [215 40 40]
tollatfel jobbra 90 előre 1 balra 90 előre :magas*70/100
töltőszín! [215 40 40] tölt
hátra :magas*70/100 jobbra 90 hátra 1 hátra :széles*32/100
balra 90 tollatle
vége
```

### 3. feladatsor: Keresztek alkalmazása

Ha a függőleges és vízszintes csíkok szimmetrikusan helyezkednek el, akkor keresztet alkotnak. Nézzük, hogy ebben az esetben milyen feladatmegoldásokkal találkozhatunk!



Alapfeladat: Svájc

Svájc lobogójának oldalaránya: 1:1. A piros szín kódja: [213 43 30] A lobogót is többféle módon megvalósíthatjuk. Kezdjük azzal a megoldással, amikor csíkokat rajzolunk!

4./a variáció: Svájc - csíkok

Svájc lobogójában két fehér csík alkalmazásával létrehozhatjuk a keresztet. A kereszt szélessége (és magassága) a lobogó szélességének 66%-a. Egy csík magassága tehát ebből kiszámítható, hiszen a teljes szélesség harmada, vagyis 22%.

Mivel a téglalaprajzoló eljárásunk úgy működik, hogy a téglalap bal alsó sarkából indul el a teknőc a rajzolás során, ügyelnünk kell arra, hogy megfelelően pozícionáljuk a teknőcöt. Az alábbi képen megjelöltük, hogy melyik téglalapot rajzoljuk ki először és annak melyik pontja lesz a rajzolásakor a bal alsó sarok. Mivel a fehér téglalap 66% széles, a két oldalon összesen 34%-nyi hely marad. Ezért a teknőccel ennek felét, tehát 17%-ot kell jobbra lépni. A csíknak pontosan középen kell elhelyezkednie. Ahhoz, hogy pontosan középre kerüljünk a magasság 50%-át kell előre lépni, de ebből le kell vonnunk a csík magasságának (22%) felét (11%), így a magasság 41%-át kell előre lépünk.



Ezek után megrajzolhatjuk a két fennmaradó négyzetet is. Kezdjük a felsővel! Ehhez a kereszt méretének harmadát (22%) kell jobbra és felfelé lépünk. Így a kékkel jelölt pontba kerülünk.



A négyzet megrajzolása után hátra kell lépünk (44%), majd az alsó négyzetet is meg kell rajzolnunk, amelynek bal alsó sarkát a zöld ponttal jelöltük. Ezen lépések után a teknőcot az eredeti pozíciójába és irányába kell állítanunk.

```
eljárás svájc :széles :magas
  téglalap :magas :széles [213 43 30]
  tollatfel jobbra 90 előre :széles*17/100 balra 90
  előre :magas*39/100
  téglalap :magas*22/100 :széles*66/100 "fehér
  tollatfel előre :magas*22/100 jobbra 90
  előre :széles*22/100 balra 90 tollatle
  téglalap :magas*22/100 :széles*22/100 "fehér
  tollatfel hátra :magas*44/100 tollatle
  téglalap :magas*22/100 :széles*22/100 "fehér
  tollatfel hátra :magas*17/100 jobbra 90
  hátra :széles*22/100 hátra :széles*17/100 balra 90 tollatle
vége
```

#### 4./b variáció: Svájc – kereszt megrajzolása és kitöltése

Ugyanez a lobogó akár úgy is megrajzolható, hogy magát a keresztet rajzoljuk meg a teknőccel, és annak belsejét töltjük ki.

Mivel a kereszt, mint motívum a variációkban látható lobogók közül többen is előfordul érdemes a kereszt megrajzolását önálló eljárásban megvalósítani. A kereszt eljárásnak lehessen megadni a keresztet befoglaló négyzet méretét, a csík vastagságát és egy színt. A kereszt rajzolását úgy valósítuk meg, hogy a teknőc pozíciója a keresztet befoglaló négyzet bal alsó sarkát jelentse.

```
eljárás kereszt :méret :csíkméret :szín :hossz
  tollszín! :szín jobbra 90 előre :hossz balra 90 tollatle
  ismétlés 4 [előre :hossz balra 90 előre :hossz jobbra 90
    előre :csíkméret jobbra 90]
  tollatfel jobbra 45 előre 10 töltőszín! :szín tölt
  hátra 10 balra 45 balra 90 előre :hossz jobbra 90
vége
```

Ezek után a svájci zászlót kirajzoló eljárásunk a következőképpen néz ki:

```
eljárás svájc_kereszt :széles :magas
  téglalap :magas :széles [213 43 30]
  tollatfel jobbra 90 előre :széles*17/100 balra 90 előre :magas*17/100
  kereszt :széles*66/100 :széles*22/100 "fehér
  tollatfel hátra :magas*17/100 jobbra 90 hátra :széles*17/100
  balra 90 tollatle
vége
```

## Elemi feladatvariációk:



Tonga



Görögország

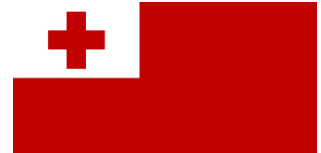


Anglia

### 1. variáció: Tonga

A kereszt eljárás felhasználásával készítsük el Tonga lobogóját is, amelynek oldalaránya 1:2! A piros szín kódja: [193 0 0]

A fehér téglalap szélessége a lobogó szélességének 42%-a, magassága a lobogó magasságának fele. A keresztet befoglaló négyzet mérete a szélesség 18%-a.



```

eljárás tonga :széles :magas
  téglalap :magas :széles [193 0 0]
  tollatfel előre :magas/2
  téglalap :magas/2 :széles*42/100 "fehér
  tollatfel előre :magas*6/100 jobbra 90 előre :széles*12/100 balra 90
  kereszt :széles*18/100 :széles*18/3/100 [193 0 0]
  tollatfel jobbra 90 hátra :széles*12/100 balra 90 hátra :magas*56/100
  tollatle
vége
    
```

### 2. variáció: Görögország

Rajzoljuk meg Görögország zászlaját is, amelynek oldalaránya: 2:3! A kék szín kódja: [13 94 175]. A csíkok magassága a lobogó magasságának 11,11%-a.

A bal felső sarokban lévő kék négyzet mérete a szélesség 37%-a.



Először rajzoljuk meg a teljes téglalapot kék színnel, majd rajzoljuk meg a csíkokat, és végül a keresztet!

```

eljárás görögország :széles :magas
  téglalap :magas :széles [13 94 175]
  ismétlés 2 [előre :magas*11,11/100
    téglalap :magas*11,11/100 :széles "fehér tollatfel
    előre :magas*11,11/100]
  tollatfel jobbra 90 előre :széles*37/100 balra 90
  ismétlés 2 [előre :magas*11,11/100
    téglalap :magas*11,11/100 :széles*63/100 "fehér tollatfel
    előre :magas*11,11/100]
  tollatfel jobbra 90 hátra :széles*37/100 balra 90 hátra :magas*4*11,11/100
  kereszt :széles*37/100 :magas*11,11/100 "fehér
vége
    
```

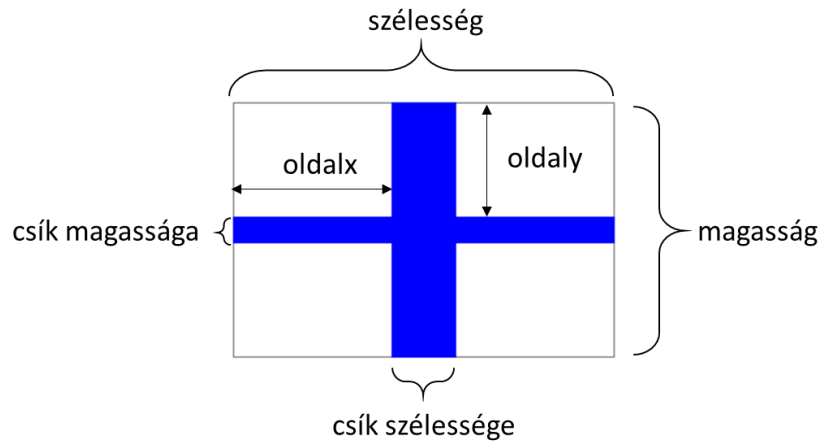
### 3. variáció: Anglia

Anglia lobogójának aránya: 3:5. A piros szín kódja: [207 20 43]. A csík szélessége a lobogó szélességének 12%-a. A variációk közül ez a lobogó



tűnik a legegyszerűbbnek. Azonban ebben az esetben a kereszt bár szimmetrikus, de nem négyzetbe, hanem téglalapba kell rajzolni. Így az eddigi kereszt rajzoló eljárásunkat nem tudjuk használni, kénytelenek vagyunk módosítani azt, hogy ne csak a kereszt szélességét, hanem magasságát is megadhassuk.

Fontos elvárás tehát, hogy az új kereszt\_téglalap eljárásnak lehessen megadni a befoglaló téglalap magasságát, szélességét, a csík magasságát és szélességét, valamint a színét.



Az adatokból ki kell számítanunk a fenti ábrán látható *oldalx* és *oldaly* távolságokat. Ezeket úgy kapjuk, hogy az téglalap szélességéből/magasságából le kell vonnunk a csík szélességét/magasságát és el kell osztanunk kettővel.

```
eljárás kereszt_téglalap :magas :széles :csíkmagas :csíkszéles :szín
  tollszín! :szín tollatle
  kereszt_tégla :magas :széles :csíkmagas :csíkszéles :szín
    (:széles-:csíkszéles)/2 (:magas-:csíkmagas)/2
vége
```

```
eljárás kereszt_tégla :magas :széles :csíkmagas :csíkszéles :szín
  :oldalx :oldaly
  tollatfel jobbra 90 előre :oldalx balra 90 tollatle
  ismétlés 2 [előre :oldaly balra 90 előre :oldalx jobbra 90
    előre :csíkmagas jobbra 90 előre :oldalx balra 90
    előre :oldaly jobbra 90 előre :csíkszéles jobbra 90]
  tollatfel jobbra 45 előre 10 töltőszín! :szín tölt
  hátra 10 balra 45 balra 90 előre :oldalx jobbra 90
vége
```

Ez alapján természetesen magának a zászlónak a kirajzolása már egyszerű:

```
eljárás anglia :széles :magas
  téglalap :magas :széles "fehér"
  kereszt_téglalap :magas :széles :széles*12/100 :széles*12/100 [207 20 43]
vége
```

#### 4. feladatsor: Csíkok és csillagok együttes alkalmazása

A lobogók között több olyat is találunk, amelyek a csíkok (téglalapok) mellett csillag alakzatot is tartalmaznak.

Alapfeladat: Kamerun

Rajzoljuk meg Kamerun zászlaját! Az oldalarány 2:3. Törekedjünk arra, hogy a zászlón látható pontos színekódokat használjuk. (zöld: [0 122 94], piros: [206 17 38], sárga: [252 208 22])



A téglalapok megrajzolásához felhasználhatjuk Belgium zászlajának megrajzolásánál használt kódot, de természetesen ki kell egészítenünk a csillag rajzolás eljárással is. A csillagok rajzolásával külön leckében foglalkoztunk. Használjuk fel az alábbi csillag rajzó eljárást:

```
eljárás csillag :n :szög :hossz :szín
  tollatle tollszín! :szín jobbra 90
  ismétlés :n [előre :hossz balra (:szög-360)/:n
              előre :hossz jobbra :szög/:n]
  tollatfel előre :hossz+1 töltőszín! :szín tölt hátra :hossz+1
vége
```

A fenti eljárás használatával egy zöld ötágú csillagot például az alábbi paraméterezéssel tudunk kirajzolni:

```
csillag 5 720 40 "zöld
```

A csillag rajzolásakor a teknőc kezdő pozíciója az alábbi ábrán látható:



Kamerun zászlajában a csillag pontosan középen helyezkedik el. A csillag oldalhossza a szélesség 6%-a. A csillag rajzolásának megkezdése előtt a teknőcot jobbra kell léptetni a szélesség 41%-ával, és felfele kell léptetni a magasság 54%-ával.

```
eljárás kamerun :széles :magas
  téglalap :magas :széles/3 [0 122 94]
  tollatfel jobbra 90 előre :széles/3 balra 90 tollatle
  téglalap :magas :széles/3 [206 17 38]
  tollatfel jobbra 90 előre :széles/3 balra 90 tollatle
  téglalap :magas :széles/3 [252 208 22]
  tollatfel jobbra 90 hátra :széles*2/3
  előre :széles*41/100 balra 90 előre :magas*54/100
  csillag 5 720 :széles*6/100 [252 208 22]
  hátra :magas*54/100 jobbra 90 hátra :széles*41/100 balra 90
vége
```

## Elemi feladatvariációk:



Panama



Mianmar



Togó



Mikronézia



Kína

### 1. variáció: Panama

Panama zászlójának oldalaránya: 2:3. Az alkalmazott színek kódjai: kék [0 82 147], piros [210 16 52]. A bal oldali csillag rajzolását a szélesség 17%-ánál, a jobb oldaliét pedig a szélesség 67%-ánál kell elkezdni. A csillagok rajzolását függőlegesen a magasság 79%-ánál, illetve 29%-ánál kell elkezdni. A csillag oldalhossza a szélesség 6%-a.



```

eljárás panama :széles :magas
  téglalaprajzol [0 82 147]
  előre :széles/2 balra 90 előre :magas/2
  téglalaprajzol [210 16 52]
  hátra :széles/2 előre :széles*17/100 balra 90
  csillagrajzol [0 82 147]
  hátra :magas*79/100 jobbra 90 előre :széles*50/100 balra 90
  csillagrajzol [210 16 52]
  hátra :magas*29/100 jobbra 90 hátra :széles*67/100 balra 90
tollatle
vége

```

```

eljárás téglalaprajzol :szín
  téglalap :magas/2 :széles/2 :szín
  tollatfel jobbra 90
vége

```

```

eljárás csillagrajzol :szín
  előre :magas*29/100
  csillag 5 720 :széles*6/100 :szín
  tollatfel
vége

```

### 2. variáció: Mianmar

Mianmar zászlójának oldalaránya: 2:3. Az alkalmazott színek kódjai: zöld [52 178 51], sárga [254 203 0], piros [234 40 57].

A csillag rajzolását a szélesség 27%-ánál, és a magasság 57%-ánál kell elkezdni. A csillag oldalhossza a lobogó szélességének 17,5%-a.





```
eljárás mianmar :széles :magas
töltőmód! 1
téglalap :magas :széles [254 203 0] ;sárga téglalap
téglalap :magas*2/3 :széles [52 178 51] ;zöld téglalap
téglalap :magas/3 :széles [234 40 57] ;piros téglalap
tollatfel előre :magas*57/100 jobbra 90 előre :széles*27/100 balra 90
csillag 5 720 :széles*17,5/100 "fehér ;csillag rajzolás
tollatfel hátra :magas*57/100 jobbra 90 hátra :széles*27/100 balra 90
töltőmód! 0
vége
```

### 3. variáció: Togó

Togó zászlójának oldalaránya: 1:1,618. Az alkalmazott színek kódjai: zöld [0 106 78], sárga [255 206 0], piros [210 16 52]. A piros négyzet oldala a lobogó magasságának 3/5-e.



A csillag rajzolását a szélesség 7%-ánál, és a magasság 75%-ánál kell elkezdeni. A csillag oldalhossza a lobogó szélességének 8,5%-a.

```
eljárás togó :széles :magas
zöldtéglalap :széles :magas
pirosnégyzet :széles :magas
fehércsillag :széles :magas
sárgacsíkok :széles :magas
vége
```

```
eljárás zöldtéglalap :széles :magas
töltőmód! 1
téglalap :magas :széles [0 106 78]
vége
```

```
eljárás pirosnégyzet :széles :magas
tollatfel előre :magas hátra :magas*3/5
téglalap :magas*3/5 :magas*3/5 [210 16 52]
vége
```

```
eljárás féhércsillag :széles :magas
tollatfel hátra :magas*2/5
jobbra 90 előre :széles*7/100 balra 90 előre :magas*75/100
csillag 5 720 :széles*8,5/100 "fehér
vége
```

```
eljárás sárgacsíkok :széles :magas
tollatfel hátra :magas*75/100 jobbra 90 hátra :széles*7/100 balra 90
tollatfel előre :magas/5
téglalap :magas/5 :széles [255 206 0]
tollatfel jobbra 90 előre :széles*37/100 balra 90 előre :magas*2/5
téglalap :magas/5 :széles*63/100 [255 206 0]
tollatfel hátra :magas*60/100 jobbra 90 hátra :széles*37/100 balra 90
tollatle töltőmód! 0
vége
```

Megjegyzés: nem véletlenül kezdtük a piros négyzet megrajzolását a bal felső saroktól visszalépve. Ezzel ugyanis el tudjuk rejteni a kerekítési hibákból fakadó problémákat. A később megrajzolt sárga csík kitakarja ugyanis a kerekítési hibából fakadó esetleges illeszkedési hibáját a piros négyzetnek.

4. variáció: Mikronézia

Mikronézia zászlaja azért különleges a variációk között, mert négy elforgatott csillagot is tartalmaz. A zászló oldalaránya sem szokványos: 10:19. A kék szín kódja:RGB kódja: [117 178 221]



Nézzük a csillagok elhelyezkedését, és azt a szöveget a vízszinteshez képest, amelyet a rajzoláskor figyelembe kell vennünk!

<b>Csillag</b>	<b>Vízszintes elhelyezkedés az szélesség százalékában</b>	<b>Függőleges elhelyezkedés a magasság százalékában</b>	<b>Szög a vízszinteshez képest</b>
bal oldali	29%	50%	-18
felső	45%	83%	0
jobb oldali	61,5%	56%	+18
alsó	45%	17%	-36

A csillag oldalhossza minden esetben a szélesség 4%-a.

```
eljárás mikronézia :széles
  kéktéglalap :széles
  balcsillag :széles
  felsőcsillag :széles
  jobbcsillag :széles
  alsócsillag :széles
vége
```

```
eljárás kéktéglalap :széles
  téglalap :széles*10/19 :széles [117 178 221]
vége
```

```
eljárás balcsillag :széles
  tollatfel előre :széles*10/19*50/100 jobbra 90 előre :széles*29/100
  balra 90 balra 18 csillag 5 720 :széles*4/100 "fehér
  tollatfel jobbra 18 hátra :széles*10/19*50/100
  jobbra 90 hátra :széles*29/100 balra 90
vége
```

```
eljárás felsőcsillag :széles
  tollatfel előre :széles*10/19*83/100 jobbra 90
  előre :széles*45/100 balra 90
  csillag 5 720 :széles*4/100 "fehér tollatfel
  hátra :széles*10/19*83/100 jobbra 90 hátra :széles*45/100 balra 90
vége
```

```
eljárás jobbcsillag :széles
  tollatfel előre :széles*10/19*56/100 jobbra 90
  előre :széles*61,5/100 balra 90
  jobbra 18 csillag 5 720 :széles*4/100 "fehér
  tollatfel balra 18 hátra :széles*10/19*56/100
  jobbra 90 hátra :széles*61,5/100 balra 90
vége
```

```

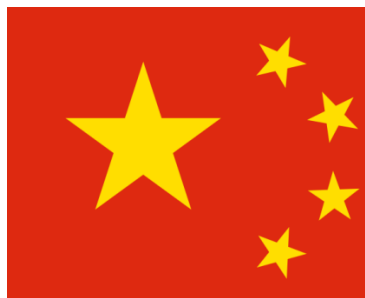
eljárás alsócsillag :széles
tollatfel előre :széles*10/19*17/100 jobbra 90
előre :széles*45/100 balra 90
balra 36 csillag 5 720 :széles*4/100 "fehér
tollatfel jobbra 36 hátra :széles*10/19*17/100
jobbra 90 hátra :széles*45/100 balra 90
vége
    
```

### 5. variáció (Kína)

Kína zászlajában is csillagok láthatóak, azonban azok szögükben és méretükben is különböznek. A lobogó oldalaránya: 2:3. A piros szín kódja: [222 41 16]. A sárga szín kódja: [255 222 0].



A lobogó egy részét felnagyítottuk, hogy jobban lássuk a csillagok elhelyezkedését.



Csillag	Vízszintes elhelyezkedés az szélesség százalékában	Függőleges elhelyezkedés a magasság százalékában	Szög a vízszinteshez képest
nagy	7%	79%	0
kicsi, felső	30,5%	87,5%	-50
kicsi, felülről második	37%	79%	-25
kicsi, felülről harmadik	37%	66%	0
kicsi, alsó	30,5%	52%	-50

A nagy csillag oldalhossza a szélesség 7,5%-a, míg a kicsiké 2,5 %.

```

eljárás kina :széles :magas
pirostéglalap :széles :magas
nagycsillag :széles :magas
kiscsillag1 :széles :magas
kiscsillag2 :széles :magas
kiscsillag3 :széles :magas
kiscsillag4 :széles :magas
vége

eljárás nagycsillag :széles :magas
tollatfel előre :magas*79/100 jobbra 90 előre :széles*7/100 balra 90
csillag 5 720 :széles*7,5/100 [255 222 0] balra 90
tollatfel hátra :magas*79/100 jobbra 90 hátra :széles*7/100 balra 90
vége
    
```

```
eljárás kiscsillag1 :széles :magas
  tollatfel előre :magas*87,5/100 jobbra 90 előre :széles*30,5/100 balra 90
  balra 50 csillag 5 720 :széles*2,5/100 [255 222 0] balra 90
  tollatfel jobbra 50 hátra :magas*87,5/100 jobbra 90
  hátra :széles*30,5/100 balra 90
vége
```

```
eljárás kiscsillag2 :széles :magas
  tollatfel előre :magas*79/100 jobbra 90 előre :széles*37/100 balra 90
  balra 25 csillag 5 720 :széles*2,5/100 [255 222 0] balra 90
  tollatfel jobbra 25 hátra :magas*79/100 jobbra 90
  hátra :széles*37/100 balra 90
vége
```

```
eljárás kicsicsillag3 :széles :magas
  tollatfel előre :magas*66/100 jobbra 90 előre :széles*37/100 balra 90
  csillag 5 720 :széles*2,5/100 [255 222 0] balra 90
  tollatfel hátra :magas*66/100 jobbra 90 hátra :széles*37/100 balra 90
vége
```

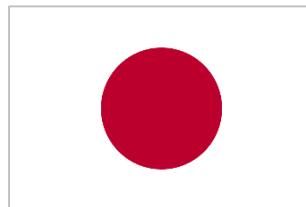
```
eljárás kicsicsillag4 :széles :magas
  tollatfel előre :magas*52/100 jobbra 90 előre :széles*30,5/100 balra 90
  balra 50 csillag 5 720 :széles*2,5/100 [255 222 0] balra 90
  tollatfel jobbra 50 hátra :magas*52/100 jobbra 90
  hátra :széles*30,5/100 balra 90
vége
```

## 5. feladatsor Körök megjelenése a zászlókon

A kör, mint síkidom, illetve azok komplexebb alkalmazásával előálló elemek is gyakran szerepelnek lobogókon.

Alapfeladat: Japán

Rajzoljuk meg Japán zászlaját! Az oldalarány 2:3. A piros szín kódja: [188 0 45]. A kör sugara a lobogó szélességének 20%-a. Hogy lássuk az eredményt a fehér lapon, szegélyt is rajzoljunk!



Először is szükségünk van olyan eljárásra, amely adott sugarú kört rajzol, a megadott színű kitöltéssel.

```
eljárás körrajzol :r :szín
  tollszín! :szín tollatfel előre :r balra 90 tollatle
  ismétlés 360 [előre :r*3,14159/180 balra 1]
  tollatfel jobbra 90 hátra :r töltőszín! :szín tölt tollatle
vége
```

```
eljárás japán :széles :magas
  téglalap :magas :széles "fehér
  szegély :magas :széles "szürke10 1
  tollatfel előre :magas/2 jobbra 90 előre :széles/2 balra 90
  körrajzol :széles*20/100 [188 0 45]
  tollatfel hátra :magas/2 jobbra 90 hátra :széles/2 balra 90 tollatle
vége
```

### Elemi feladatvariációk:



Niger



Törökország



Grönland

#### 1. variáció: Niger

Niger lobogójának oldalaránya: 6:7. A sárga szín kódja: [224 85 6]. A zöld színé: [13 176 43]. A kör sugara a zászló szélességének 12%-a.

```
eljárás niger :széles :magas
  téglalap :magas :széles [224 85 6]
  téglalap :magas*2/3 :széles "fehér"
  téglalap :magas/3 :széles [13 176 43]
  tollatfel előre :magas/2 jobbra 90 előre :széles/2 balra 90
  körrajzol :széles*12/100 [224 85 6]
  tollatfel hátra :magas/2 jobbra 90 hátra :széles/2 balra 90 tollatle
vége
```

#### 2. variáció: Törökország

Törökország lobogója látszólag kilóg a sorból, hiszen itt egy holdsarlót kell megvalósítani. Azonban ha tüzetesebben megvizsgáljuk a feladatot, rájöhethetünk, hogy a holdsarlót megvalósíthatjuk egy fehér kitöltésű körrel, amelyet részben takar egy piros kör.



Az oldalarány ebben az esetben 2:3, a piros szín kódja: [227 10 23].

A fehér kör bal oldalának széle a szélesség 18,5%-ára esik, és az 52,5%-ig tart. Vagyis az átmérő a szélesség 35%-a, így a sugár 17%. A piros kör bal oldali pontja a lobogó szélességének 27%-ra esik. Az átmérője a szélesség 27%-a. A csillag bal oldali széle a szélesség 49%-án kezdődik, oldalhossza 6%. Mind a holdsarló, mind a csillag függőlegesen középre van igazítva.

```
eljárás törökország :széles :magas
  téglalaprajzolás :széles :magas
  fehércör :széles :magas
  piroskör :széles :magas
  csillagrajzolás :széles :magas
vége
```

```
eljárás téglalaprajzolás :széles :magas
  töltőmód! 1
  téglalap :magas :széles [227 10 23]
vége
```

```
eljárás fehércör :széles :magas
  tollatfel előre :magas/2 jobbra 90 előre :széles*35,5/100 balra 90
  körrajzol :széles*17/100 "fehér"
  tollatfel hátra :magas/2 jobbra 90 hátra :széles*35,5/100 balra 90
  tollatle
vége
```

```

eljárás piroskör :széles :magas
  tollatfel előre :magas/2 jobbra 90 előre :széles*40,5/100 balra 90
  körrajzol :széles*13,5/100 [227 10 23]
  tollatfel hátra :magas/2 jobbra 90 hátra :széles*40,5/100 balra 90
  tollatle
vége

```

```

eljárás csillagrajzolás :széles :magas
  tollatfel előre :magas/2 jobbra 90 előre :széles*49/100 balra 90
  balra 18 csillag 5 720 :széles*6/100 "fehér
  tollatfel jobbra 18 hátra :magas/2 jobbra 90 hátra :széles*49/100
  balra 90 tollatle töltőmód! 0
vége

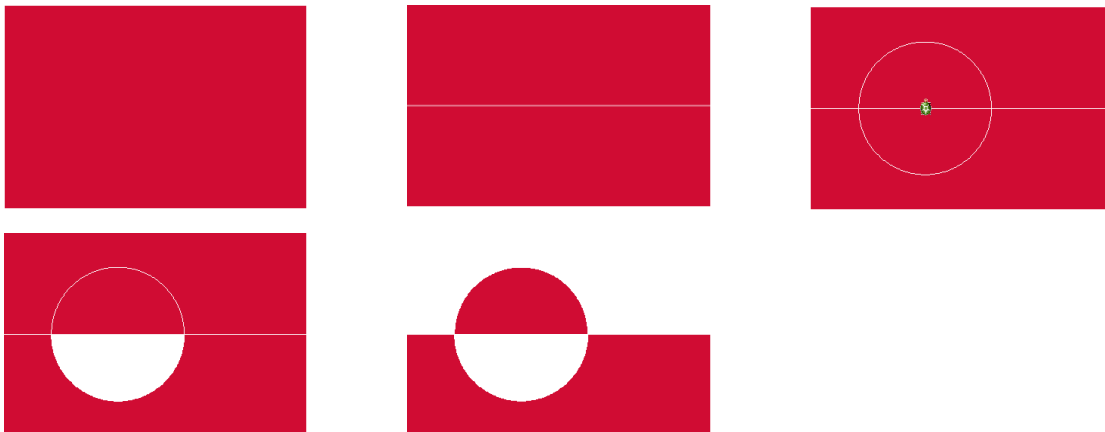
```

### 3. variáció: Grönland

Grönland lobogójának oldalárányai: 2:3. A bordó szín kódja: [208 12 51]. A kör középpontja 38%-ra esik, sugara 22%.



A megvalósítás többféle lehet, első ránézésre akár félkörök színezésével is meg lehetne oldani a feladatot, ehhez azonban szükség lenne a félkör eljárásra is. Mi egy olyan megoldást mutatunk, amelyben a meglévő körrajzol eljárást használjuk fel. Először a teljes zászlót kitöltjük piros színnel, majd a magasság felénél megrajzolunk egy fehér, vízszintes vonalat. Ezután megrajzoljuk a fehér kört és az alsó félkört kitöltjük fehérrel, a felső téglalapot pedig fehér színnel. Vagyis a lépések:



```

Eljárás grönland :széles :magas
  bordótéglalap :széles :magas
  fehérvonal :széles :magas
  félkörkitöltés :széles :magas
  fehértéglalap :széles :magas
vége

```

```

eljárás bordótéglalap :széles :magas
  töltőmód! 1
  téglalap :magas :széles [208 12 51]
vége

```

```

eljárás fehérvonal :széles :magas
  előre :magas/2 tollszín! "fehér jobbra 90 előre :széles
  tollatfel hátra :széles balra 90 hátra :magas/2
vége

```

```
eljárás félkörkitöltés :széles :magas
előre :magas/2 jobbra 90 előre :széles*38/100 balra 90 tollatle
körrajzol :széles*22/100 "fehér
tollatfel hátra 10 töltőszín! "fehér tölt előre 10
hátra :magas/2 jobbra 90 hátra :széles*38/100 balra 90
vége
```

```
eljárás fehértéglalap :széles :magas
előre (:magas/2)+1 töltőszín! "fehér tölt
hátra (:magas/2)+1 töltőmód! 0 tollatle
vége
```

**Kérdés:** A fehér kör rajzolásánál miért csak a körvonal látszik, miért nem lett kitöltve fehérrel a teljes kör?

## 6. feladatsor: Háromszögek

A háromszögekkel is gyakran találkozhatunk a nemzetek lobogóin. Először valósítsuk meg a Bahama-szigetek zászlóját, amely a csíkok mellett egy fekete háromszöget is tartalmaz!

Alapfeladat: Bahama-szigetek



A zászló oldalaránya 1:2. A színek kódjai: kék [0 171 201], sárga [250 224 66]. A szabályos háromszög oldalhossza a lobogó magasságával egyenlő. A belső szögei 60 fokokak, ez alapján kell kiszámolnunk a háromszög megrajzolásához szükséges elfordulások szögeit.

```
eljárás bahama :széles :magas
töltőmód! 1
téglalap :magas :széles [0 171 201] ;teljes terület kék téglalap
téglalap :magas*2/3 :széles [249 227 0] ;sárga téglalap
téglalap :magas/3 :széles [0 171 201] ;kék téglalap
tollszín! "fekete ;háromszög megrajzolása
jobbra 60 előre :magas balra 120 előre :magas jobbra 60 hátra :magas
tollatfel jobbra 45 előre 10 tölt ;háromszög kitöltése
hátra 10 balra 45 tollatle töltőmód! 0
vége
```

### Elemi feladatvariációk:



Seychelle-szigetek



Bahrein

#### 1. variáció: Seychelle-szigetek

A Seychelle-szigetek lobogóját sok háromszög alkotja, de valójában itt nem kell háromszögeket megrajzolnunk, szakaszokkal kell felosztanunk a lobogót, és az így kapott területeket kell kifesteni. A lobogó oldalaránya: 1:2. A színek kódjai: kék [0 63 135], sárga [252 216 86], piros [214 40 40], zöld [0 122 61].

A megvalósításnál problémát okozhat, hogy bár a teknőccel való elfordulást könnyen megvalósítjuk, ki kellene számolnunk azt is, hogy mekkora szakaszt kell rajzolnunk ahhoz, hogy a szakasz ne essen kívülre a lobogót tartalmazó téglalap területéből. Ezt persze megtehetnénk, azonban ennél a példánál egy másik módszert mutatunk be, amelyet kisebbek is sikerrel alkalmazhatnak a feladatmegoldás során. A módszer abból áll, hogy menjünk az átlós szakaszok végére, majd az `xypoz!` paranccsal pozícionáljuk a teknőcöt a kiindulópontra.

A kék háromszög jobb felső csúcsa a lobogó szélességének harmadára esik. A sárgáé a kétharmadára. A piros terület alsó szakasza a zászló magasságának kétharmadáig tart, a zöld pedig az egyharmadáig.

```
tanuld seychelles :széles :magas
fekete :széles :magas xpoz ypoz irány sárga :széles :magas xpoz ypoz irány
piros xpoz ypoz :széles :magas irány zöld xpoz ypoz irány :széles :magas
vége
```

```
tanuld fekete :széles :magas :x :y :i
tollszín! 0 töltőszín! 0 előre :magas jobbra 90 előre :széles/3 balra 90
xypoz! :x :y irány! :i
tollatfel jobbra 10 előre :magas/2 tölt hátra :magas/2 balra 10 tollatle
vége
```

```
tanuld sárga :széles :magas :x :y :i
tollszín! 14 töltőszín! 14 tollatfel előre :magas jobbra 90
előre :széles/3 tollatle előre :széles/3 balra 90
xypoz! :x :y irány! :i
tollatfel jobbra 35 előre :magas/2 tölt hátra :magas/2 balra 35 tollatle
vége
```

```
tanuld piros :széles :magas :x :y :i
tollszín! 12 töltőszín! 12 tollatfel előre :magas jobbra 90
előre :széles*2/3 tollatle előre :széles/3 balra 90 hátra :magas/3
xypoz! :x :y irány! :i
tollatfel jobbra 60 előre :magas/2 tölt hátra :magas/2 balra 60 tollatle
vége
```

```
tanuld zöld :széles :magas :x :y :i
tollszín! 2 töltőszín! 2 jobbra 90 előre :széles balra 90 előre :magas/3
xypoz! :x :y irány! :i
tollatfel jobbra 85 előre :magas/2 tölt hátra :magas/2 balra 85 tollatle
vége
```

## 2. variáció: Bahrein

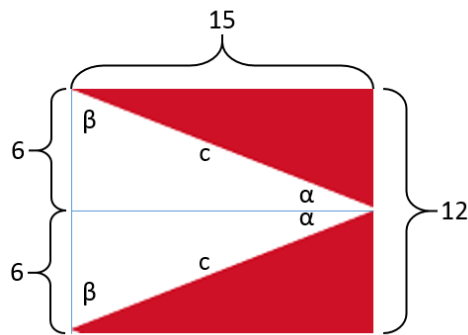
Nézzünk olyan példát is, ahol a háromszögek kirajzolásánál a szögfüggvényeket is alkalmazzuk!

Bahrein lobogóján egy háromszögekből álló cikk-cakk vonal látható. A lobogó oldalaránya: 3:5. A piros szín kódja: [206 17 38]. A cikk-cakk vonal szélessége (vagyis egy fehér (vagy piros ☺) háromszög magassága) a lobogó szélességének 15%-a. Maga a cikk-cakk vonal a zászló szélességének negyedénél kezdődik. Egy fehér háromszög alapja könnyen kiszámolható, hiszen 5 darab ilyen alkotja a zászlót, így egy háromszög alapja a lobogó magasságának egyötöde. Ha a lobogó szélessége 100 képpont, akkor a lobogó magassága  $100 \cdot 3/5 = 60$ . Így egy fehér háromszög alapja  $60/5 = 12$ .





Hogyan számolhatjuk ki a háromszög oldalhosszát, illetve a szögeket?



Tudjuk, hogy  $\text{tg } \alpha = 6/15 = 0,4$ . Az alfa szög tehát  $\arctan 0,4$  érték lesz, ami hozzávetőleg 21,8 fokot jelent. A  $\beta$  szög  $90-\alpha$  képlettel kiszámítható.

Az oldal hosszát akár Pitagorasz tétellel is kiszámíthatnánk, de maradjunk most a szögfüggvényeknél.

$c = 15/\cos \alpha = 15 / \cos 21,8$ . Ez az érték hozzávetőleg 16,15.

Természetesen a pontos értékek kiszámítását a számítógépre bizzuk. Ez alapján az eljárásunk:

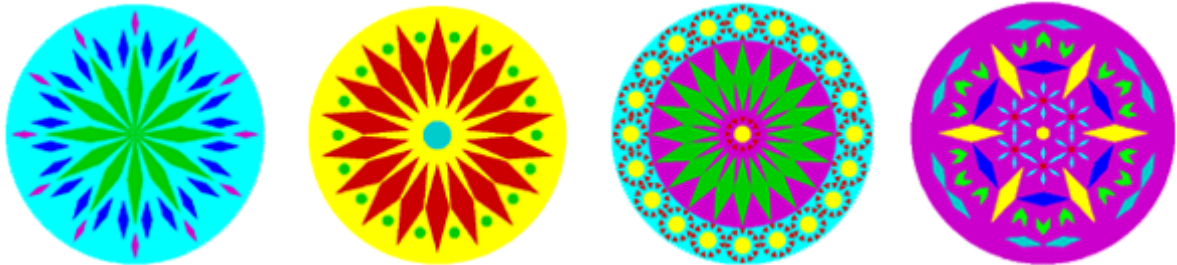
```

eljárás bahrein :széles :magas
  szegély :magas :széles "szürke10 1
  tollatfel jobbra 90 előre :széles/4 balra 90 tollszín! [206 17 38]
  tollatle jobbra (90-arctan (:magas/5/2)/(:széles*15/100))
  ismétlés 5 [előre ((:széles*15/100)/
    cos arctan (:magas/5/2)/(:széles*15/100))
    balra 180-2*arctan (:magas/5/2)/(:széles*15/100)
    előre ((:széles*15/100)/
    cos arctan (:magas/5/2)/(:széles*15/100))
    jobbra 180-2*(arctan (:magas/5/2)/(:széles*15/100))]
  tollatfel balra (90-arctan (:magas/5/2)/(:széles*15/100))
  hátra :magas/2 jobbra 90 előre :széles/2
  töltőszín! [206 17 38] tölt
  hátra :széles*3/4 balra 90 hátra :magas/2
vége
  
```

## Térkitöltés forgatással

A térkitöltést (mozaikot) klasszikus esetben sormintából kiindulva készítjük el. Egy ettől teljesen különböző eset az, amikor a területet egy körcikkekbe rajzolt ábra forgatásával fedjük le. A körcikkek lehetnek diszjunktak, illetve átfedőek is.

Nézzünk néhány példát (<http://teamlabor.inf.elte.hu/logosecsetvonasok/lecke3.html>):

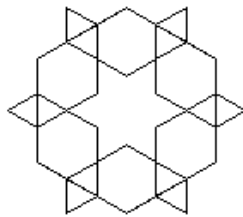


### Forgatott elemek

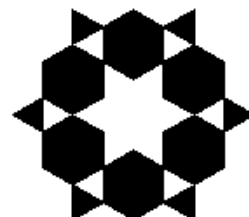
Az egyik megvizsgálandó eset, amikor a mozaik többféle elemből áll. Ennek alelete, amikor az egyes elemek egymás elforgatottjai.

### Jégvirág

A befagyott ablakokon gyakran láthatók jégvirágok. Az alábbi két ábra egy jégvirág határvonalait, illetve magát a jégvirágot mutatja. Az ábrákon látható minden kis szakasz egyforma hosszú. Készíts két eljárást (határ :hossz, illetve jégvirág :hossz), amely megrajzolja ezeket az ábrákat!



határ 10



jégvirág 10

A határvonalak felváltva hatszögek és háromszögek, a csillagot és a háromszögek párjait – amint a színezett ábrán látható – nem kell megrajzolni!

```
eljárás határ :hossz
```

```
  ismétlés 6 [ismétlés 6 [előre :hossz jobbra 60] előre :hossz  
    ismétlés 3 [előre :hossz balra 120]  
    előre :hossz*2 jobbra 60]
```

```
vége
```

```
eljárás jégvirág :hossz
```

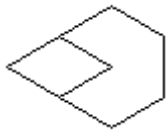
```
  ismétlés 6 [ismétlés 6 [előre :hossz jobbra 60]  
    tollatfel jobbra 60 előre 2 tollatle tölt hátra 2  
    balra 60 előre :hossz  
    ismétlés 3 [előre :hossz balra 120]  
    tollatfel balra 30 előre 2 tollatle tölt hátra 2  
    jobbra 30 előre :hossz*2 jobbra 60]
```

```
vége
```

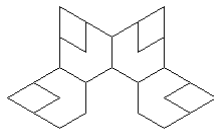
### Virágablak

Virágokból érdekes mintákat állíthatunk össze.

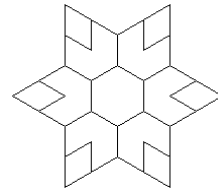
Készíts Logo eljárást (virágablak :x :db), amely :db mennyiségű :x méretű virágmintát rajzol a képernyőre!



virágablak 40 1



virágablak 20 4



virágablak 20 6

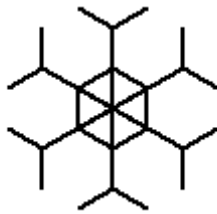
eljárás virágablak :h :db

```
ismétlés :db [előre :h balra 60 előre :h balra 60 előre 2*:h balra 120
előre :h hátra :h jobbra 120 hátra :h balra 120 előre :h
jobbra 120 előre :h balra 120 előre :h balra 60 előre :h
balra 60 előre :h jobbra 60]
```

vége

### Jégvirág – 1. változat

Itt a tél! Készítsd el a jégvirág1 :hossz és a jégvirág2 :hossz eljárásokat, melyek az alábbi hópolyheket rajzolják:



jégvirág1 10



jégvirág2 10

Kétféleképpen állhatunk a feladat megoldásához:

- hat körcikket rajzolunk, a körcikk tartalmát egyetlen eljárásba írjuk;
- egymásra helyezünk több forgatott ábrát.

Az első néhány esetben a második megoldást választjuk. A jégvirág1 egy hatszögből és hat elforgatott elemből áll.

eljárás jégvirág1 :hossz

```
külső :hossz előre :hossz jobbra 120 belső :hossz balra 120 hátra :hossz
vége
```

eljárás külső :hossz

```
ismétlés 6 [ág :hossz jobbra 60]
vége
```

eljárás belső :hossz

```
ismétlés 6 [előre :hossz jobbra 60]
vége
```

eljárás ág: hossz

```
előre 2*:hossz balra 60 vonal :hossz jobbra 120
vonul :hossz balra 60 hátra 2*:hossz
vége
```

eljárás vonal : hossz

```
előre :hossz hátra :hossz
vége
```

Itt az előző ábra hatszögének oldalait egy-egy háromszög két oldalával helyettesítjük.

```
eljárás jégvirág2 :hossz
  külső2 :hossz előre :hossz jobbra 60
  belső2 :hossz jobbra 120 hátra :hossz
vége
```

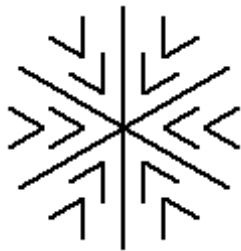
```
eljárás külső2 :hossz
  ismétlés 6 [ág2 :hossz jobbra 60]
vége
```

```
eljárás belső2 :hossz
  ismétlés 6 [ismétlés 2 [előre :hossz jobbra 120] jobbra 180]
vége
```

```
eljárás ág2: hossz
  előre 2*:hossz balra 60
  ismétlés 3 [vonal :hossz jobbra 60]
  balra 120 hátra 2*:hossz
vége
```

### Jégvirág – 2. változat

Itt a tél! Készítsd el a jégvirág3 :hossz és a jégvirág4 :hossz eljárásokat, melyek az alábbi hópehelyeket rajzolják!



jégvirág3 10



jégvirág4 10

A jégvirág hatos szimmetriájú. Az egyes részekben is ismétlődnek a V alakú elemek. Itt is kétféleképpen állhatunk a feladat megoldásához, most az első lehetőséget választjuk:

- hat körcikket rajzolunk, a körcikk tartalmát egyetlen eljárásba írjuk;
- egymásra helyezünk több forgatott ábrát.

```
eljárás jégvirág3 :hossz
  ismétlés 6 [cikk3 :hossz jobbra 60]
vége
```

```
eljárás cikk3 :hossz
  vonal 3*:hossz jobbra 30 tollatfel
  ismétlés 2 [előre :hossz balra 30 tollatle vonal :hossz
             jobbra 60 vonal :hossz balra 30 tollatfel]
  hátra 2*:hossz balra 30 tollatle
vége
```

```
eljárás jégvirág4 :hossz
  ismétlés 6 [cikk4 :hossz jobbra 60]
vége
```

```

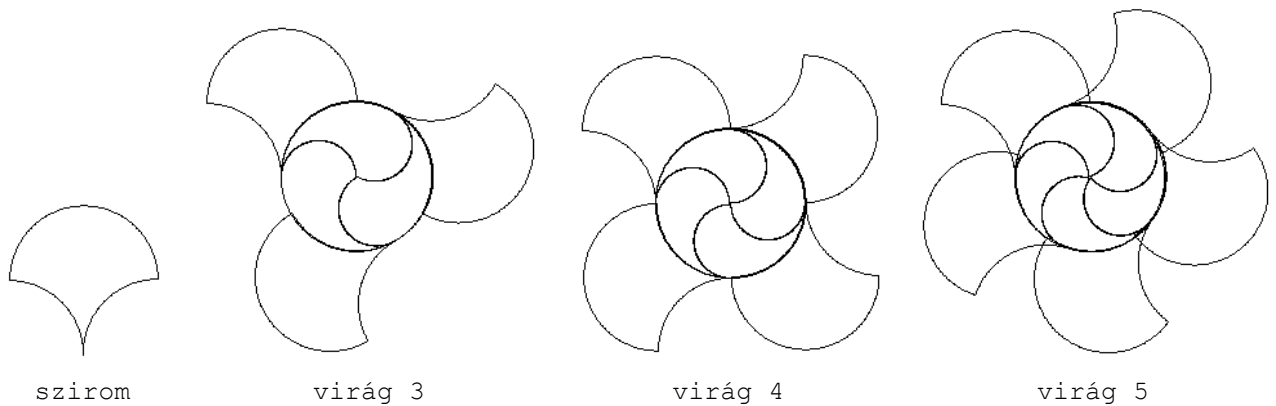
eljárás cikk4: :hossz
  előre 2*:hossz balra 60
  ismétlés 3 [vonal :hossz jobbra 60]
  balra 120 hátra 2*:hossz jobbra 30 tollatfel
  ismétlés 2 [előre :hossz balra 30 tollatle vonal :hossz
              jobbra 60 vonal :hossz balra 30 tollatfel]
  hátra 2*:hossz balra 30 tollatle előre :hossz jobbra 120
  vonal :hossz balra 120 hátra :hossz
vége

```

### Virágszirmok

Egy virágszirmot körívek határolnak, mint az alábbi ábrán látható. A szirmok egyik körívükkel illeszkednek a kör alakú virág belső részre, át is fedhetik egymást. A belső részen is megjelenik olyan körívekből álló mintázat, amelyben a rajzolási lépéshosszak fele akkorák, mint, a szirmoknál használt lépéshossz.

Készíts Logo eljárást szirm (szirm), valamint :n darab szirmot tartalmazó virág rajzolására (virág :n)!



Az előzőekkel szemben itt a körívek száma (azaz a körívek szöge) nem rögzített, hanem paraméterként kapjuk. Az első megoldás alapján a két ábrát (szirm, illetve a virág belseje) külön rajzoljuk.

```

eljárás virág :db
  ismétlés :db [szirm 1 jkörív 360/:db 1]
  ismétlés :db [jkörív 180 0,5 jobbra 180 bkörív 180 0,5
                jobbra 180 jkörív 360/:db 1]
vége

eljárás szirm :h
  bkörív 90 :h jobbra 90 jkörív 180 :h jobbra 90 bkörív 90 :h jobbra 180
vége

eljárás bkörív :szög :h
  ismétlés :szög [előre :h balra 1]
vége

eljárás jkörív :szög :h
  ismétlés :szög [előre :h jobbra 1]
vége

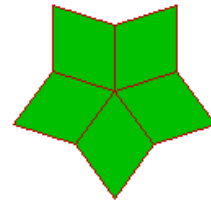
```

## Rombuszok

Készíts eljárásokat az alábbi ábrák rajzolására (rombusz :h, belső :h, külső :h, rombuszok :h), ahol :h a rombusz oldalhossza!

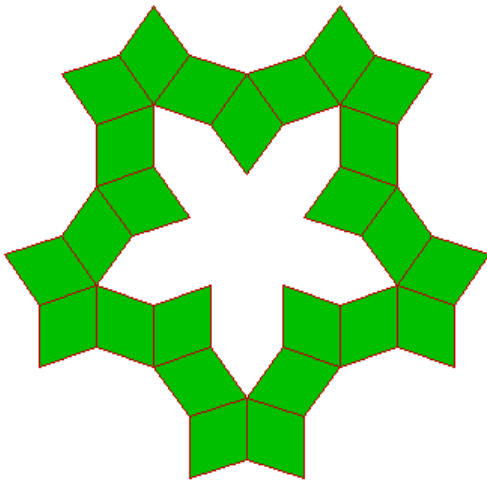


rombusz 100

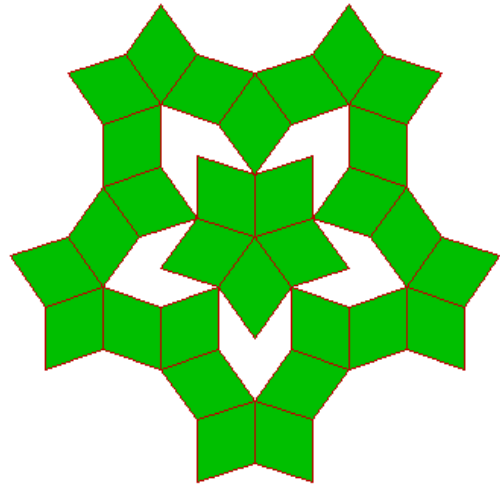


belső 90

A belső ábra 5 rombusz, 72 fokkal elforgatva egymáshoz képest. Érdekes külön megrajzolni.

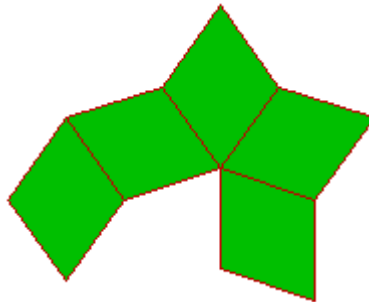


külső 80



rombuszok 80

A külső ábra megrajzolásában is 5 ismétlődő rész van, azt kell alkalmasan kiválasztani és alapként megrajzolni:



```

eljárás alap :h
  tollatfel előre :h balra 36 tollatle rombusz :h
  előre :h jobbra 72 előre :h jobbra 36 rombusz :h
  előre :h jobbra 72 előre :h balra 180 rombusz :h
  ismétlés 2 [jobbra 72 rombusz :h] jobbra 144 tollatfel
  ismétlés 3 [előre :h balra 36] jobbra 36 jobbra 180 tollatle
vége

```

```

eljárás belső :h
  ismétlés 5 [rombusz :h jobbra 72]
vége

```

```
eljárás külső :h
  ismétlés 5 [alap :h jobbra 72]
vége
```

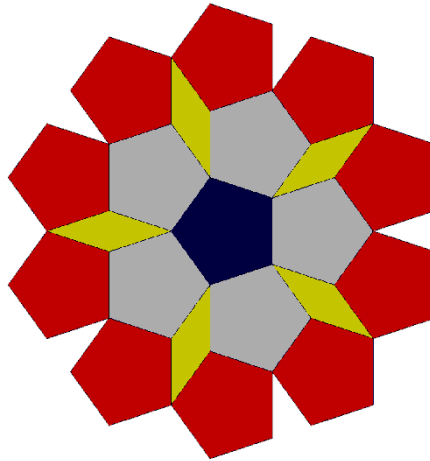
```
eljárás rombusz :h
  ismétlés 2 [előre :h jobbra 72 előre :h jobbra 108]
  tollatfel jobbra 36 előre :h/2 tölt hátra :h/2 balra 36 tollatle
vége
```

```
eljárás rombuszok :h
  belső :h külső :h
vége
```

### Penrose-mozaik – 1. változat

Az alábbi, Penrose-mozaik színes ötszögekből készül úgy, hogy a közöttük kimaradó négyszög alakú területeket is beszínezzük.

Készíts Logo eljárást (`penrose :h`), amely `:h` oldalhosszúságú ötszögekből a mellékelt ábrán látható Penrose-mozaikot készíti el!



Az ábra egy kék ötszög, amit szürke ötszögek vesznek körbe. A szürke ötszögek két oldalára két piros ötszöget kell rajzolni. A sárga rombuszokat nem kell külön megrajzolni, elég az összes ötszög megrajzolása után befesteni.

```
eljárás penrose :h
  balra 36 ötszög :h [0 0 64]
  ismétlés 5 [szürke :h előre :h jobbra 72]
  ismétlés 5 [sárga :h jobbra 72] tollatle
vége
```

```
eljárás piros :h
  balra 36 ötszög :h [192 0 0] jobbra 36
vége
```

```
eljárás szürke :h
  balra 108 ötszög :h [172 172 172]
  előre :h piros :h jobbra 72 előre :h piros :h jobbra 72
  ismétlés 3 [előre :h jobbra 72]
  jobbra 108
vége
```

```
eljárás sárga :h
  töltőszín! [196 196 0] tollatfel
  balra 120 előre :h tölt hátra :h jobbra 120 előre :h tollatle
vége
```

```
eljárás ötszög :h :s
  ismétlés 5 [előre :h jobbra 72]
  tollatfel jobbra 36 előre :h/2 töltőszín! :s tölt hátra :h/2
  balra 36 tollatle
vége
```

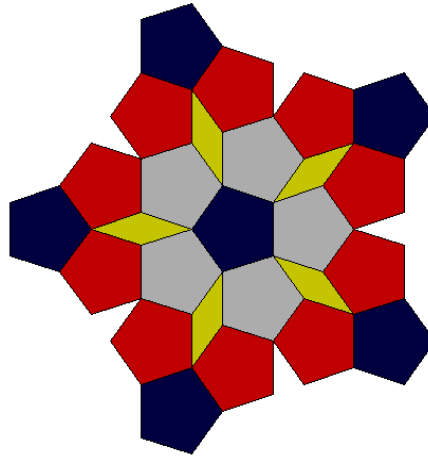
### Penrose-mozaik – 2. változat

Az alábbi, Penrose-mozaik színes ötszögekből készül úgy, hogy a közöttük kimaradó négyszög alakú területeket is beszínezzük.

Készíts Logo eljárást (`penrose :h`), amely `:h` oldalhosszúságú ötszögekből a mellékelt ábrán látható Penrose-mozaikot készíti el!

A feladat az előző bővítése, a külső piros ötszögpárok egy-egy kék ötszöget fognak közre. Többféle elképzelés is lehet a megoldásra:

- az egyik piros ötszög rajzolás meghív megfelelő helyen egy kék ötszög rajzoló eljárást;
- a sárga rombusz festéshez illesztjük be a kék ötszög rajzolást;
- harmadik forgatott ábraként képzeljük az újabb kék ötszögeket.



Most válasszuk a harmadik megoldási lehetőséget!

```
eljárás penrose :h
  balra 36 ötszög :h [0 0 64]
  ismétlés 5 [szürke :h előre :h jobbra 72]
  ismétlés 5 [sárga :h jobbra 72]
  ismétlés 5 [balra 144 előre :h jobbra 36 előre :h balra 72
    kék :h jobbra 72 hátra :h balra 36 hátra :h
    jobbra 144 előre :h jobbra 72]
vége
```

```
eljárás szürke :h
  balra 108 ötszög :h [172 172 172]
  előre :h piros :h jobbra 72 előre :h piros :h jobbra 72
  ismétlés 3 [előre :h jobbra 72]
  jobbra 108
vége
```

```
eljárás kék :h
  ötszög :h [0 0 64]
vége
```

```
eljárás piros :h
  balra 36 ötszög :h [192 0 0] jobbra 36
vége
```



```
eljárás sárga :h
  töltőszín! [196 196 0] tollatfel
  balra 120 előre :h tölt hátra :h jobbra 120 előre :h tollatle
vége
```

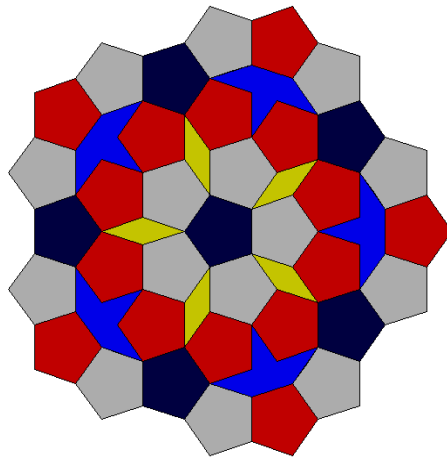
```
eljárás ötszög :h :s
  ismétlés 5 [előre :h jobbra 72]
  tollatfel jobbra 36 előre :h/2 töltőszín! :s tölt
  hátra :h/2 balra 36 tollatle
vége
```

### Penrose-mozaik – 3. változat

Az alábbi, Penrose-mozaik színes ötszögekből készül úgy, hogy a közöttük kimaradó nem ötszög alakú területeket is beszínezzük.

Készíts Logo eljárást (`penrose :h`), amely `:h` oldalhosszúságú ötszögekből a mellékelt ábrán látható Penrose-mozaikot készíti el!

Itt kívül újabb ötszögek (piros és szürke) jelennek meg, valamint egy papírból hajtogatott csónakra emlékeztető világosabb kék sokszög.



```
eljárás penrose :h
  balra 36 ötszög :h [0 0 64]
  ismétlés 5 [szürke :h előre :h jobbra 72]
  ismétlés 5 [sárga :h jobbra 72]
  ismétlés 5 [balra 144 előre :h jobbra 36 előre :h balra 72
    kék :h jobbra 72 hátra :h balra 36 hátra :h
    jobbra 144 előre :h jobbra 72]
vége
```

```
eljárás ötszög :h :s
  ismétlés 5 [előre :h jobbra 72]
  tollatfel jobbra 36 előre :h/2 töltőszín! :s tölt
  hátra :h/2 balra 36 tollatle
vége
```

```
eljárás szürke :h
  balra 108 ötszög :h [172 172 172]
  előre :h piros :h jobbra 72 előre :h piros :h jobbra 72
  ismétlés 3 [előre :h jobbra 72]
  jobbra 108
vége
```

Most a kék ötszög rajzoló hívja meg a külső szürke és piros ötszögeket rajzoló eljárásokat.

```

eljárás kék :h
  ötszög :h [0 0 64] jobbra 108
  előre :h balra 72 ötszög :h [172 172 172] jobbra 108
  előre :h balra 72 ötszög :h [192 0 0] jobbra 108
  előre :h balra 72 ötszög :h [172 172 172] jobbra 108
  balra 36 hátra :h balra 36 hátra :h/2
  tollatfel jobbra 90 előre :h/4 töltőszín! [0 0 232] tölt
  hátra :h/4 balra 90 tollatle
  hátra :h/2 balra 36 hátra :h balra 108
vége
  
```

```

eljárás sárga :h
  töltőszín! [196 196 0] tollatfel
  balra 120 előre :h tölt hátra :h jobbra 120 előre :h tollatle
vége
  
```

```

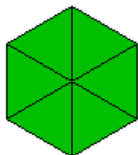
eljárás piros :h
  balra 36 ötszög :h [192 0 0] jobbra 36
vége
  
```

### Hatszögek

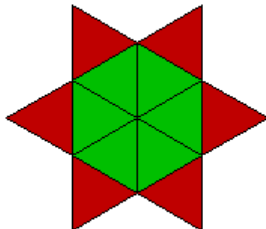
Készíts Logo eljárásokat (hár :oldal :szín, hat1 :oldal, hat2 :oldal, hat3 :oldal) az alábbi háromszögekből álló ábrák rajzolására, ahol :oldal a háromszög oldalhossza! A belső hat háromszög zöld, a körülöttük levő hat piros, a legkülső tizenkettő pedig kék színnel legyen kifestve



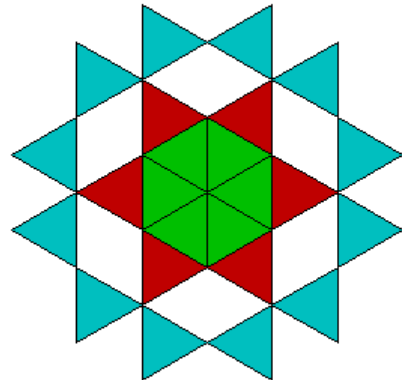
hár 50 2



hat1 50



hat2 50



hat3 50

Hatszög alapúak a rajzok, azaz mindegyiknél hat körcikkbe kell rajzolni a megfelelő elemeket.

```

eljárás hár :h :sz
  ismétlés 3 [előre :h balra 120]
  balra 30 tollatfel előre :h/2 töltőszín! :sz tölt hátra :h/2
  tollatle jobbra 30
vége
  
```

```

eljárás hat1 :h
  ismétlés 6 [hár :h 2 balra 60]
vége
  
```

A második ábrán minden zöld háromszögre illesztünk egy-egy piros háromszöget – a megoldásban vastagon szedtük a bővítést.

```

eljárás hat2 :h
  ismétlés 6 [hár :h 2 előre :h balra 60 háromszög :h 4
              jobbra 60 hátra :h balra 60]
vége
  
```

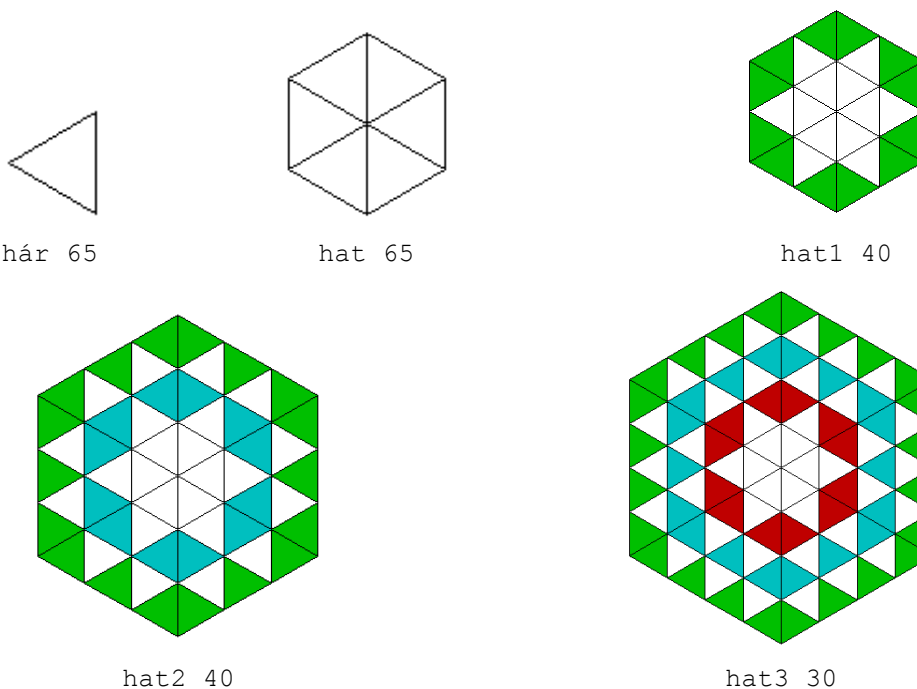
A harmadik ábrán minden piros háromszög külső csúcsába rajzolunk két-két kék háromszöget – a megoldásban most is vastagon szedtük a bővítést.

```

eljárás hat3 :h
  ismétlés 6 [hár :h 2 előre :h balra 60 háromszög :h 4
              előre :h háromszög :h 3 jobbra 120 háromszög :h 3
              balra 120 hátra :h jobbra 60 hátra :h balra 60]
vége
  
```

### Háromszögek

Írj Logo eljárásokat (hár :oldal, hat :oldal, hat1 :oldal, hat2 :oldal, hat3 :oldal) az alábbi háromszögekből álló ábrák rajzolására, ahol :oldal a háromszög oldalhossza! A külső háromszögek legyenek zöld, az eggyel beljebb levők kék, a még beljebb levők pedig piros színűek! A legbelső 6 háromszög mindig legyen festetlen!



A feladat érdekessége, hogy a korábbiaktól eltérően másképp is megfogható. Egy olyan mozaikot kell rajzolnunk, ami kifelé bővül. Az elemek mérete nem változik az egyes körökben, de az elemek száma igen.

```

eljárás hár :h :sz
  ismétlés 3 [előre :h balra 120]
  balra 30 tollatfel előre :h/2 töltőszín! :sz tölt hátra :h/2
  tollatle jobbra 30
vége
  
```

```

eljárás hat :h
  ismétlés 6 [hár :h 15 balra 60]
vége
  
```

```

eljárás hat1 :h
  ismétlés 6 [hár :h 15 előre :h nő :h 2
              balra 120 előre :h jobbra 120 nő :h 2
              balra 60 hátra :h]
vége

```

Egy háromszög mindenképpen van mindegyik hatod körcikkbën, a további háromszögeket pedig a kifelé haladás sorszámától függően a nő eljárás rajzolja.

```

eljárás nő :h :sz
  háromszög :h :sz előre :h
  ha :sz>2 [nő :h :sz-1]
  balra 120 előre :h jobbra 120
  ha :sz>2 [nő :h :sz-1]
  balra 60 hátra :h jobbra 60
vége

```

```

eljárás hat2 :h
  ismétlés 6 [hár :h 15 előre :h nő :h 3
              balra 120 előre :h jobbra 120 nő :h 3
              balra 60 hátra :h]
vége

```

```

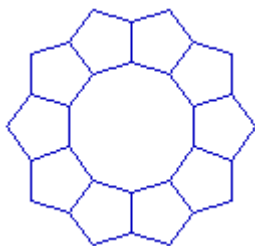
eljárás hat3 :h
  ismétlés 6 [hár :h 11 előre :h nő :h 4
              balra 120 előre :h jobbra 120 nő :h 4
              balra 60 hátra :h]
vége

```

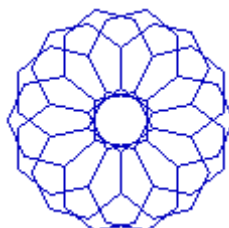
## Virág

Szabályos sokszögekből (legalább 5 oldalúból) úgy készíthetünk virágot, hogy egymás mellé helyezzünk belőlük annyit, hogy éppen körbeérjenek. Ez a legtöbb esetben nem jön ki pontosan, de ha kétszer annyi sokszöget rajzolunk, mint ahány oldalú a sokszög, akkor biztosan elkészül az ábra.

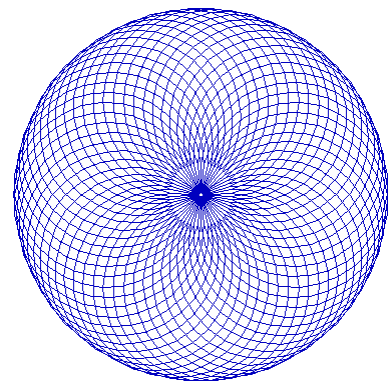
Készíts Logo eljárást (`virág :n :h`), amely `:n` oldalú, `:h` oldalhosszúságú sokszögekből virágot készít!



virág 5 20



virág 7 20



virág 29 5

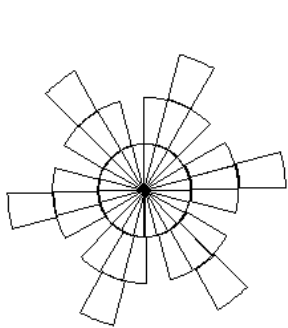
```

eljárás virág :n :h
  ismétlés 2* :n [ismétlés :n [előre :h jobbra 360/:n]
                 balra 360/:n hátra :h jobbra 180-360/:n]
vége

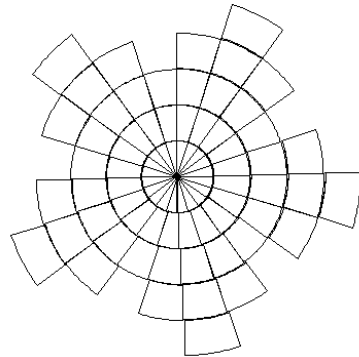
```

## Mozaik

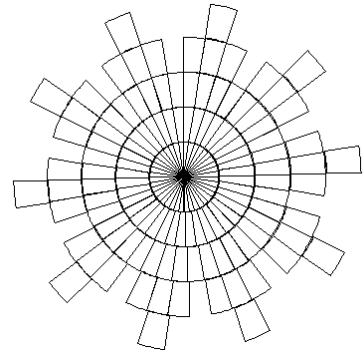
Készíts körmozaikot (mozaik :db :táv :hány), amely :db darab :táv szélességű körgyűrűt rajzol! A mozaik két legkülső gyűrűjében már nem minden alapelem látszik! A :hány a legkülső gyűrűben látható alapelemek száma legyen!



mozaik 3 25 6



mozaik 5 25 5



mozaik 5 25 10

A két külső gyűrűt külön rajzoljuk meg, majd a mozaik belsejét egy rekurzív eljárással.

```

eljárás mozaik :db :táv :hány
  elsőgyűrű :db :táv :hány
  tollatfel jobbra 90 előre :táv balra 90 tollatle
  másodikgyűrű :db :táv :hány
  tollatfel jobbra 90 előre :táv balra 90 tollatle
  mozaikbelső :db-2 :táv :hány
vége

eljárás elsőgyűrű :db :táv :hány
  ismétlés :hány [tollatfel körív 360/(:hány*4) :db*:táv tollatle
    berajzol :táv körív 360/(:hány*4) :db*:táv
    berajzol :táv tollatfel
    ismétlés 2 [körív 360/(:hány*4) :db*:táv]]
vége

eljárás másodikgyűrű :db :táv :hány
  ismétlés :hány [ismétlés 3 [berajzol :táv
    körív 360/(:hány*4) (:db-1)*:táv
    berajzol :táv]
    tollatfel körív 360/(:hány*4) (:db-1)*:táv]
vége

eljárás mozaikbelső :db :táv :hány
  ha :db>0 [ismétlés :hány*4 [körív 360/(:hány*4) :db*:táv berajzol :táv]
    jobbra 90 előre :táv balra 90 mozaikbelső :db-1 :táv :hány]
vége

eljárás berajzol :táv
  jobbra 90 előre :táv hátra :táv balra 90
vége

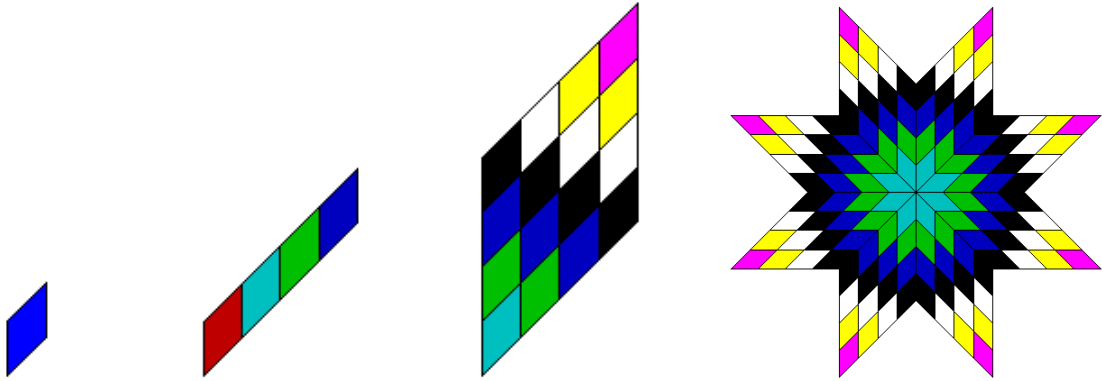
eljárás körív :szög :r
  ismétlés :szög [előre 2*3,14159*:r/360 jobbra 1]
vége

```

## Forgó

Egy mozaikot színes rombuszokból építünk össze (rombusz :h :s), amelynek oldalhossza :h, kisebbik szöge pedig :s fokos. A sor :m :h :s eljárás :m darab, különböző színű rombuszból álló sort rajzol, a mozaik :n :m :h :s eljárás pedig :n darab sort helyez egymás

mellé úgy, hogy a rombuszok színe átlósan egyforma legyen! Legvégül pedig az fmozaik :n :m :h :s eljárás a mozaikot elforgatja :s fokonként, amíg körbe nem érünk.



rombusz 10  
45

sor 4 5 45

mozaik 4 4 5 45

fmozaik 4 4 5 45

```
eljárás rombusz :h :s
  ismétlés 2 [előre :h jobbra :s előre :h jobbra 180-:s]
  tollatfel jobbra :s/2 előre :h tölt hátra :h balra :s/2 tollatle
vége
```

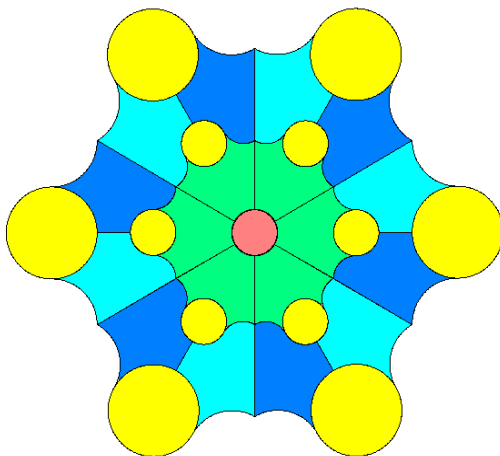
```
eljárás sor :n :h :s
  töltőszín! maradék töltőszín+15 16 rombusz :h :s
  ha :n>1 [jobbra :s előre :h balra :s sor :n-1 :h :s jobbra :s
  hátra :h balra :s]
vége
```

```
eljárás mozaik :n :m :h :s
  töltőszín! maradék :n 15 sor :m :h :s
  ha :n>1 [előre :h mozaik :n-1 :m :h :s hátra :h]
vége
```

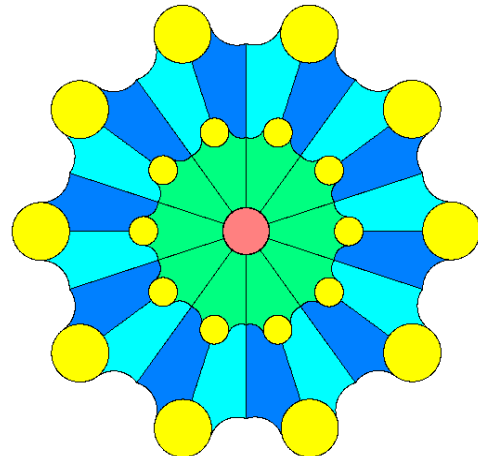
```
eljárás fmozaik :n :m :h :s
  ismétlés 360/:s [mozaik :n :m :h :s jobbra :s]
vége
```

### Díszüveg

Készíts Logo eljárást (üveg :n :r) :n cikkből álló ablaküveg készítésére, ahol :r a cikkek sugara (a leghosszabb szakasz végpontjának távolsága a piros kör középpontjától)!



üveg 6 100



üveg 10 100

```
eljárás üveg :n :h
  ismétlés :n [elemi :h 360/:n]
vége
```

Az :sz szöget bezáró T hosszú szakaszok végpontjának távolsága:  $2 \cdot T \cdot \sin$  :sz/2

```
eljárás elemi :h :sz
  tollatfel előre :h/8 tollatle előre :h*7/8 tollatfel hátra :h
  tollatle jobbra :sz tollatfel előre :h/8 tollatle
  előre :h*7/8 balra 90+:sz/2
  jobbív 90 :h/2*sin :sz/2 balív 540 :h/2*sin :sz/2 körtölt 90
  jobbív 90 :h/2*sin :sz/2
  balra 90+:sz/2 előre :h/2 balra 90-:sz/2
  balív 90 :h/4*sin :sz/2 jobbív 90 :h/4*sin :sz/2
  balra 90 egyenes jobbra 90
  jobbív 450 :h/4*sin :sz/2 körtölt -90 balív 90 :h/4*sin :sz/2
  balra 90-:sz/2 hátra :h*3/8 balra 90 balív 360 :h/8
  színezés :h :sz
vége
```

```
eljárás balív :f :r
  ismétlés :f [előre :r*3,14159/180 balra 1]
vége
```

```
eljárás jobbív :f :r
  ismétlés :f [előre :r*3,14159/180 jobbra 1]
vége
```

```
eljárás körtölt :ir
  tollatfel töltőszín! [255 255 0] balra :ir előre 2 tölt
  hátra 2 jobbra :ir tollatle
vége
```

```
eljárás egyenes
  tollatfel előre 1
  ha pontszín = 15 [egyenes]
  tollatle hátra 1
vége
```

```
színezés :h :sz
  tollatfel jobbra 90 hátra :h/8
  töltőszín! [0 128 255] balra :sz/4 előre :h tölt hátra :h
  töltőszín! [0 255 128] balra :sz/4 előre :h/4 tölt hátra :h/4
  töltőszín! [0 255 255] balra :sz/4 előre :h tölt hátra :h
  jobbra 3*:sz/4 töltőszín! [255 128 128] tölt tollatle
vége
```

## Sorminták

Sorminta alatt egy vonal mentén ismétlődő díszítő elemeket értjük. Hol találkozhatunk velük? Eleget csak körülnéznünk és mindenhol sormintákat látunk: a terítő szélén, a párnákon, a függönyökön, a tapéta bordúrnén, szalvétákon és sorolhatnánk még hosszan. A népművészetben a sorminták jellegzetesek az egyes tájegységekre, de a világ különböző népei is más-más díszítéseket alkalmaznak, amelyekből a hozzáértők egyetlen pillantással áthatják, hogy honnan valók.

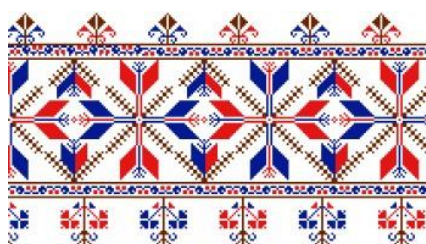
Nézzünk néhány példát (a képek a wikipediáról származnak):



Kalotaszegi



Tapéta bordúr



Orosháza



Szalvéta dísz



Marton



Terítő széle

Már az elsősök is készítenek különböző sormintákat a füzetükbe. Most megtanuljuk, hogy lehet számítógéppel, Imagine Logo-val egyszerűbb sormintákat készíteni.

### Sorminta azonos elemekből

#### Írásos hímzés

Készítsd el az ábra szerint a mintát (minta :méret), illetve a sormintát kirajzoló sorminta :db :méret eljárást, ahol a :db a sormintában előforduló minták számát, a :méret pedig a nagyságát befolyásolja!



minta 50



sorminta 3 50

eljárás minta :oldal

előre 2\*:oldal jobbra 90 előre 4\*:oldal balra 90

előre :oldal balra 90 előre :oldal

balra 90 előre :oldal\*2 jobbra 90 előre 2\*:oldal balra 90

előre :oldal balra 90 előre :oldal

balra 90 előre :oldal\*4 jobbra 90 előre 2\*:oldal

tollatfel hátra 2\*:oldal balra 90 hátra 2\*:oldal jobbra 90

hátra 2\*:oldal balra 90 hátra 2\*:oldal tollatle

vége



A sormintában a már elkészített mintát használjuk fel. Gondoljunk arra, hogy a sorminta kirajzolása után vissza kell térni az indulási pozícióba!

```

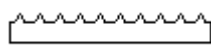
eljárás sorminta :db :méret
  ismétlés :db [minta :méret tollatfel előre 2*:méret jobbra 90
                előre 2*:méret balra 90 előre 2*:méret jobbra 90
                előre 2*:méret balra 90 tollatle]

  tollatfel
  ismétlés :db [jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret
                jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret]

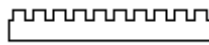
  tollatle
vége
  
```

### Fűrész

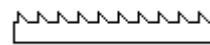
Készíts 3 eljárást, amely az alábbi fűrészfog-mintákat képes rajzolni! Az egyes eljárások paramétere a fűrészfogak száma legyen!



afűrész 10



bfűrész 10



cfűrész 10

```

eljárás Afog :h
  előre :h/4 balra 60 előre :h/2 jobbra 120
  előre :h/2 balra 60 előre :h/4
vége

eljárás afűrész :db
  jobbra 90 ismétlés :db [Afog 100/:db]
  jobbra 90 előre 100/:db jobbra 90 előre 100 jobbra 90 előre 100/:db
vége

eljárás Bfog :h
  előre :h/4 balra 90 ismétlés 3 [előre :h/2 jobbra 90]
  balra 180 előre :h/4
vége

eljárás bfűrész :db
  jobbra 90 ismétlés :db [Bfog 100/:db]
  jobbra 90 előre 100/:db jobbra 90 előre 100 jobbra 90 előre 100/:db
vége

eljárás Cfog :h
  előre :h/4 balra 90 előre :h/2 jobbra 135
  előre :h/2*gyök 2 balra 45 előre :h/4
vége

eljárás cfűrész :db
  jobbra 90 ismétlés :db [Cfog 100/:db]
  jobbra 90 előre 100/:db jobbra 90 előre 100 jobbra 90 előre 100/:db
vége
  
```

### Nyomok

Egy esős nap után Peti beszaladt a szobába sáros lábbal. Készíts olyan nyom :hossz és nyomok :db :hossz eljárásokat, amelyek az ábrának megfelelően kirajzolja a lábnyomokat! A :db paraméter a lábnyomok számát, a :hossz paraméter pedig a méretét jelöli.



nyom 50



nyomok 5 50

Érdeemes figyelni arra, hogy áttekinthető legyen a kód! Készítsünk rész elemeket!

```

eljárás nyom :méret
  tollatfel jobbra 90 előre 2*:méret/5
  balra 90 előre :méret/10 jobbra 180 tollatle
  talprész :méret/4-:méret/10 :méret/5 "igaz
  tollatfel jobbra 90 előre 2*:méret/5
  jobbra 90 előre :méret/4 tollatle
  talprész :méret/2-:méret/10 :méret/5 "igaz
  tollatfel hátra :méret/2 tollatle
vége

eljárás talprész :méret :sugár :teli
  előre :méret félkör :sugár előre :méret jobbra 90
  előre 2*:sugár jobbra 90
  ha :teli [tollatfel jobbra 45 előre :sugár/2 tölt
            hátra :sugár/2 balra 45 tollatle]
vége

eljárás félkör :sugár
  ismétlés 180 [előre :sugár*3,14*2/360 jobbra 1]
vége

eljárás alap :méret :tele
  tollatfel
  ismétlés 4 [előre :méret jobbra 90]
  tollatfel jobbra 90 előre :méret/4 balra 30 előre :méret/8 balra 90
  tollatle
  nyom :méret/2
  tollatfel jobbra 90 hátra :méret/8 jobbra 30
  előre :méret/4 balra 90 előre :méret/2 jobbra 30 tollatle
  nyom :méret/2
  tollatfel balra 30 hátra :méret/2 jobbra 90
  hátra :méret/4 hátra :méret/4 balra 90 tollatle
vége

```

Ha elkészültünk egy nyommal, már egyszerű sorozat rajzolásról van szó.

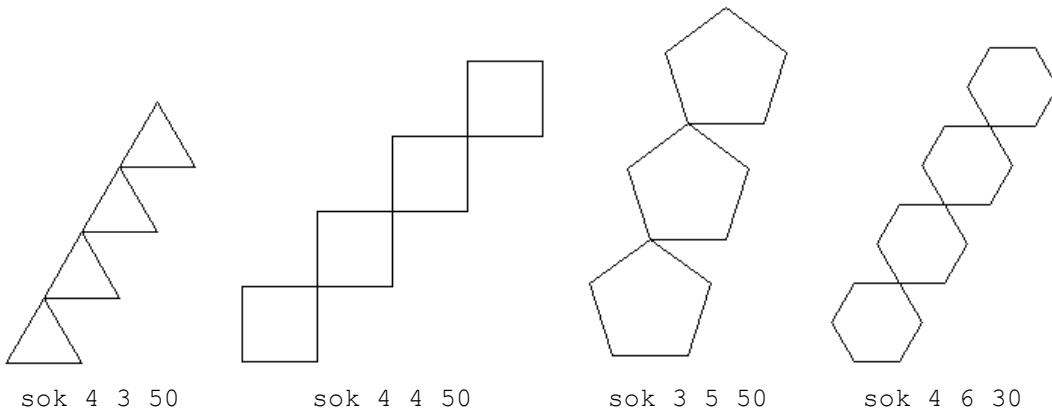
```

eljárás nyomok :db :méret
  ismétlés :db [alap :méret "igaz tollatfel előre :méret tollatle]
  tollatfel hátra :db*:méret tollatle
vége

```

### Sokszögek

Készíts Logo eljárást (sok :db :n :h), amely :db darab :n oldalú szabályos sokszöget rajzol az alábbi ábráknak megfelelő elrendezésben! A sokszögek oldalhossza legyen :h!



A feladatban a sokszögekhez tartozó szögek kiszámítása okozhat csak nehézséget! Figyeljünk arra, hogy páratlan csúcsú sokszögeknél is a kiindulási csúccsal szemközti csúcsba érjünk!

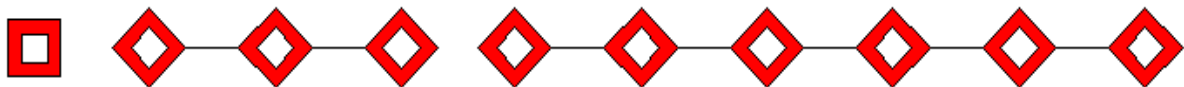
```

eljárás sok :db :n :h
  balra 90-360/:n
  ismétlés :db [ismétlés egészshányados :n*3 2 [előre :h jobbra 360/:n]
                balra 180 ha maradék :n 2 = 1 [jobbra 180/:n]]
  tollatfel
  ismétlés :db [ha maradék :n 2 = 1 [balra 180/:n] balra 180
                ismétlés egészshányados :n*3 2 [balra 360/:n hátra :h]]
  jobbra 90-360/:n tollatle
vége
  
```

## Sorminta egy alapelemmel és más záró elemmel

### Nyaklánc

Gyöngyökből láncot fűzünk. Ehhez készítsd el a gyöngy eljárást! A gyöngyöket fessd színesre! Írd meg a lánc :n eljárást!

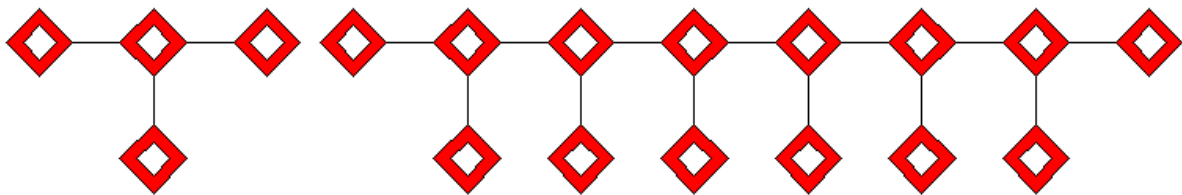


Gyöngy

lánc 3

lánc 6

Hasonlóan készítünk díszes láncot is. Írd meg a díszes :n eljárást!



díszes 3

díszes 8

```

eljárás gyöngy
  töltőszín! "piros
  ismétlés 4 [előre 30 jobbra 90]
  tollatfel jobbra 45 előre 30/4*gyök 2 balra 45 tollatle
  ismétlés 4 [előre 15 jobbra 90]
  tollatfel balra 135 előre 30/8*gyök 2 tölt
  előre 30/8*gyök 2 jobbra 135 tollatle
vége
  
```

```

eljárás lánc :db
  jobbra 90
  ismétlés :db-1 [balra 45 gyöngy ismétlés 2 [előre 30 jobbra 90]
                balra 135 előre 30]
  balra 45 gyöngy jobbra 45
  tollatfel hátra (:db-1)*30*(1+gyök 2) tollatle
  balra 90
vége
  
```

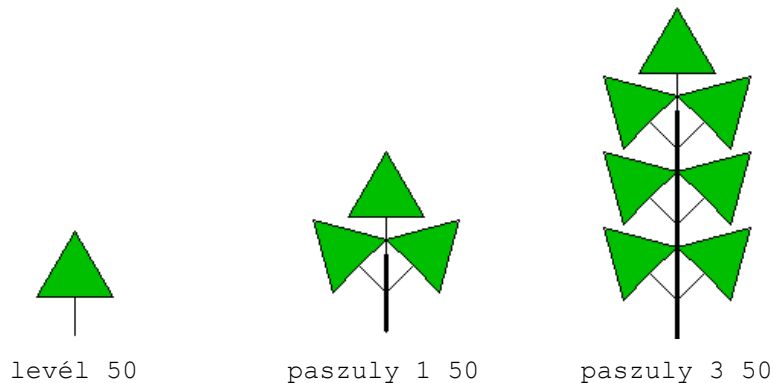
Külön rajzoljuk meg a két szélső gyöngyöt, a középsőket sorként adhatjuk meg.

```

eljárás díszes :db
  jobbra 45 gyöngy
  ismétlés 2 [előre 30 jobbra 90] balra 135 előre 30
  ismétlés :db-2 [balra 45 gyöngy ismétlés 3 [előre 30 jobbra 90]
    balra 135 előre 30 balra 45 gyöngy balra 135
    előre 30 jobbra 45 előre 30 jobbra 45 előre 30]
  balra 45 gyöngy jobbra 45
  tollatfel hátra (:db-1)*30*(1+gyök 2) tollatle
  balra 90
vége
  
```

### Égig érő paszuly

Rajzold le a mesebeli égig érő paszulyt! Készítsd el először az ábra szerinti levél eljárást (levél :méret), ahol a levelet egy szabályos háromszögből állíthatod elő, ahol a :méret a háromszög oldalhossza! Színezd is ki! Írd meg a paszuly rajzoló eljárást is (paszuly :db :méret), ahol a :db paraméter a levélpárok számát adja meg, a :méret paraméter pedig a levél nagyságát és a levélpárok távolságát határozza meg! A paszuly szára legyen háromszor olyan vastag, mint a levél szára!



```

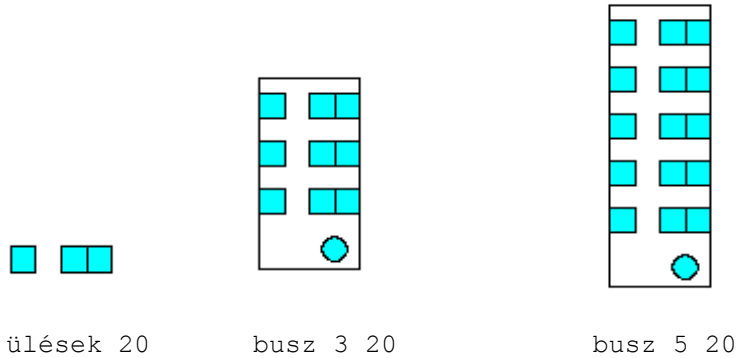
eljárás paszuly :n :r
  tollvastagság! 3 előre :n*:r hátra :n*:r tollvastagság! 1
  ismétlés :n [előre :r/2 jobbra 45 levél :r balra 90 levél :r
    jobbra 45 előre :r/2]
  levél :r hátra :n*:r
vége
  
```

```

eljárás levél :r
  előre :r/2 balra 90 előre :r/2 jobbra 120
  ismétlés 2 [előre :r jobbra 120]
  előre :r/2 jobbra 90
  tollatfel töltőszín! "zöld előre 3 tölt hátra 3 tollatle
  hátra :r/2
vége
  
```

### Autóbusz

Figyeld meg a mellékelt rajzokat és készíts olyan eljárásokat, amely kirajzol egy ülősort (ülések :méret) és egy másikat, ami magát az autóbuszt rajzolja ki (busz :sordb :méret)! A :sordb paraméter a buszban lévő ülősorok számát adja meg! Ne feledkezz meg a vezetőről sem, aki a dupla ülések előtt ül! Színezd is ki a rajzodat!



```

eljárás ülések :méret
  tollatfel előre :méret/2 tollatle
  ismétlés 4 [előre :méret/4 jobbra 90]
  tollatfel jobbra 45 előre 3 tölt hátra 3 balra 45 tollatle
  tollatfel jobbra 90 előre :méret/2 balra 90 tollatle
  ismétlés 2 [ismétlés 4 [előre :méret/4 jobbra 90] tollatfel
    jobbra 45 előre 3 tölt hátra 3 balra 45 tollatle
    jobbra 90 előre :méret/4 balra 90]
  tollatfel jobbra 90 hátra :méret balra 90 tollatle
vége
  
```

Először kirajzoljuk a busz körvonalát, majd a vezetőt és üléseket is elhelyezzük.

```

eljárás busz :n :méret
  töltőszín! "ibolya
  ismétlés 2 [előre (:n+1)*:méret/2 jobbra 90 előre :méret jobbra 90]
  vezető :méret
  ismétlés :n [ülések :méret]
  tollatfel hátra :n*:méret/2 tollatle
vége
  
```

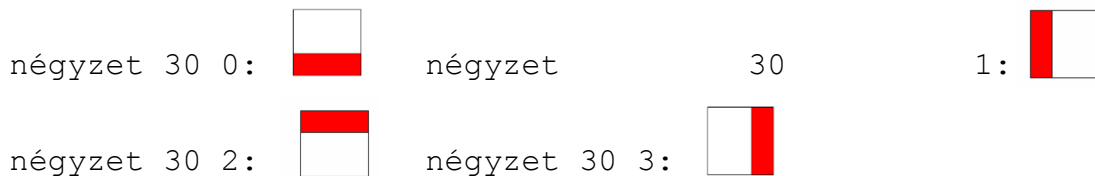
```

eljárás vezető :méret
  tollatfel előre :méret/4 jobbra 90 előre :méret/4*3 tollatle
  kör :méret/4 tölt
  tollatfel hátra :méret/4*3 balra 90 hátra :méret/4 tollatle
vége
  
```

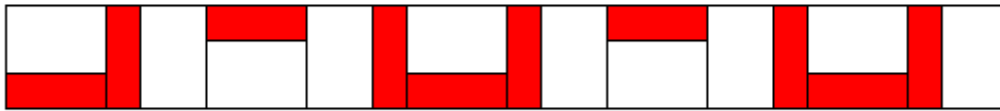
## Sorminta elforgatott alapelemmel

### Sorminta

Készíts sormintát rajzoló eljárást (sorminta :n :oldal :melyik)! A sorminta :n darab :oldal hosszúságú négyzetből áll. A négyzetekbe rendre az előzőhöz képest az óramutató járása szerint 90 fokkal elforgatott színes csíkot rajzolj (a :melyik paraméter adja meg, hogy melyik a legelső négyzet)! A négyzet :oldal :melyik eljárás rajzolja az alapelemet, ami a következőképpen nézhet ki:



A sorminta:



sorminta 10 50 0

```

eljárás négyzet :oldal :melyik
  ismétlés 4 [előre :oldal jobbra 90]
  ismétlés :melyik [előre :oldal jobbra 90]
  ismétlés 2 [előre :oldal/3 jobbra 90 előre :oldal jobbra 90]
  tollatfel jobbra 45 előre 5 tölt hátra 5 balra 45 tollatle
  ismétlés 4-:melyik [előre :oldal jobbra 90]
vége

eljárás sorminta :n :oldal :melyik
  ha :n>0 [négyzet :oldal maradék :melyik 4 jobbra 90 előre :oldal
    balra 90 sorminta :n-1 :oldal :melyik+1 tollatfel
    jobbra 90 hátra :oldal balra 90 tollatle]
vége

```

## Sorminta két alapelemből

### Hullámok

Egy hullámvonalat labdákkal díszítettünk az alábbi módon. Készíts Logo eljárásokat (kívül :db :arány, belül :db :arány), amelyek a díszített hullámvonalakat rajzolják! Az első paraméter azt adja meg, hogy hány hullámhegy legyen a rajzon, a második pedig azt, hogy a labda ke-rülete hányadrésze a hullámhegyet alkotó félkör hosszának!

Vegyük észre, hogy negyed körökből érdemes felépíteni az eljárásokat! Ezek lehetnek balra vagy jobbra ívelők. Figyeljünk arra, hogy a kirajzolás után vissza kell térni az eredeti pozícióba! Ehhez észre kell venni, hogy páros vagy páratlan helyzetben mit kell csinálni!

```

eljárás belül :db :arány
  ismétlés :db [bnkör :arány*2 ismétlés 4 [bnkör :arány]
    bnkör :arány*2 jnkör :arány*2
    ismétlés 4 [jnkör :arány] jnkör :arány*2]
  tollatfel jobbra 180
  ha maradék :db 2=1 [bnkör :arány*2 bnkör :arány*2 jnkör :arány*2
    jnkör :arány*2]
  ismétlés 2*egészhányados :db 2 [bnkör :arány*2 bnkör :arány*2
    jnkör :arány*2 jnkör :arány*2]
  jobbra 180 tollatle
vége

eljárás kívül :db :arány
  ismétlés :db [bnkör :arány*2 ismétlés 4 [jnkör :arány]
    bnkör :arány*2 jnkör :arány*2
    ismétlés 4 [bnkör :arány] jnkör :arány*2]
  tollatfel
  jobbra 180 ha maradék :db 2=1 [bnkör :arány*2 bnkör :arány*2
    jnkör :arány*2 jnkör :arány*2]
  ismétlés 2*egészhányados :db 2 [bnkör :arány*2 bnkör :arány*2
    jnkör :arány*2 jnkör :arány*2]
  jobbra 180 tollatle
vége

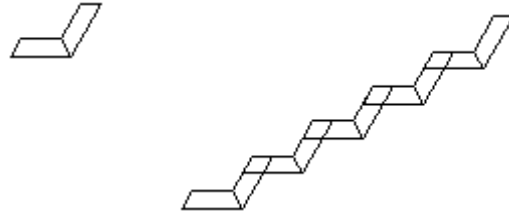
eljárás bnkör :méret
  ismétlés 9 [előre :méret balra 10]
vége

```

```
eljárás jnkör :méret
  ismétlés 9 [előre :méret jobbra 10]
vége
```

### Lépcső

Készíts Logo programot, amely az ábrán látható alapelemet tudja rajzolni! Két alapelemekből egy lépcsőfokot épít, illetve a lépcsőfokokat egymáshoz illeszti! A három eljárás neve és paraméterei az alábbiak legyenek: alap :a :b (ahol :b a trapéz oldalának a hossza, a :a pedig a trapéz párhuzamos oldalai közül a rövidebb hossza, a hosszabbik hossza ekkor :a+:b, a szögek pedig 60, illetve 120 fokosak), fok :a :b, lépcső :db :a :b!



alap 20 10 fok 20 10 lépcső 5 20 10

```
eljárás alap :a :b
  előre :b jobbra 60 előre :a jobbra 60
  előre :b jobbra 120 előre :a+:b jobbra 120
vége
```

```
eljárás fok :a :b
  alap :a :b jobbra 60 előre :a+:b balra 120
  alap :a :b előre :b jobbra 60 előre :a-:b
vége
```

A lépcsőnél is térjünk vissza a kezdő pozícióba!

```
eljárás lépcső :db :a :b
  jobbra 30 ismétlés :db [fok :a :b]
  tollatfel jobbra 180
  ismétlés :db [fok :a :b] jobbra 180 balra 30 tollatle
vége
```

### Zongora

Egy zongorán fekete és fehér billentyűk vannak. A fekete billentyűk 2-es és 3-as csoportokban helyezkednek el.

Készíts Logo eljárást (zongora :db), amely :db 2-es és 3-as csoportot tartalmazó zongorabil-  
lentyűzetet rajzol!



A zongora kirajzolásához készítsük el a különböző típusú billentyűket!

```
eljárás zongora :db
  ismétlés :db [hármás négyes]
  tollatfel jobbra 90 hátra :db*5*14 balra 90 tollatle
vége
```

eljárás hármás

jobbfehér fekete kettősfehér fekete balfehér  
 jobbra 90 előre 10 balra 90

vége

eljárás négyes

jobbfehér fekete kettősfehér fekete kettősfehér fekete balfehér  
 jobbra 90 előre 10 balra 90

vége

eljárás fekete

jobbra 90 előre 10 balra 90 előre 20 balra 90 előre 3 jobbra 90  
 ismétlés 6 [előre 30 hátra 30 jobbra 90 előre 1 balra 90]  
 jobbra 90 hátra 3 balra 90 hátra 20

vége

eljárás jobbfehér

előre 50 jobbra 90 előre 7 jobbra 90 előre 30 balra 90 előre 3  
 jobbra 90 előre 20 jobbra 90 előre 10 jobbra 90

vége

eljárás kettősfehér

előre 20 jobbra 90 előre 3 balra 90 előre 30 jobbra 90 előre 4  
 jobbra 90 előre 30 balra 90 előre 3 jobbra 90 előre 20  
 jobbra 90 előre 10 jobbra 90

vége

eljárás balfehér

előre 20 jobbra 90 előre 3 balra 90 előre 30 jobbra 90 előre 7  
 jobbra 90 előre 50 jobbra 90 előre 10 jobbra 90

vége

## Sorminta több alapelemből, záró elemmel

### Nyaklánc

Kétféle színű gyöngyből láncot fűzünk. Készítsd el a gyöngy :szín eljárást! A gyöngyöket ferd színesre a :szín paraméterben meghatározott színnel! Írd meg a lánc :n eljárást ilyen láncok rajzolására!

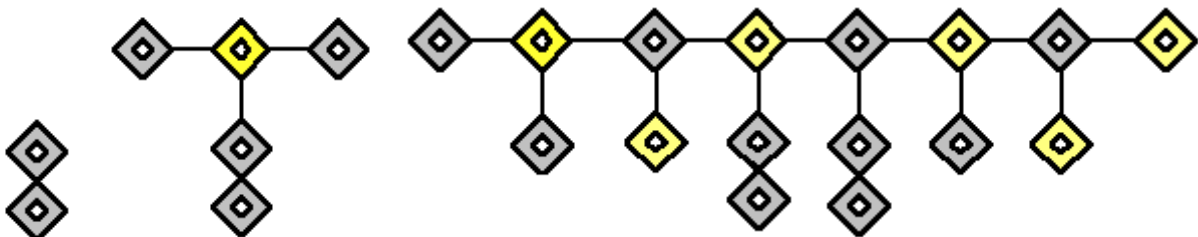


gyöngy "szürke" gyöngy "sárga"

lánc 3

lánc 6

Készítünk díszített láncot is! A díszített lánc középső – vagy két középső – szemére ékkő kerül. Írd meg az ékkő és a díszített :n eljárást is!



ékkő

díszített 3

díszített 8



```
eljárás gyöngy :szín
töltőszín! :szín
ismétlés 4 [előre 30 jobbra 90]
tollatfel jobbra 45 előre 30/4*gyök 2 balra 45 tollatle
ismétlés 4 [előre 15 jobbra 90]
tollatfel balra 135 előre 30/8*gyök 2 tölt
előre 30/8*gyök 2 jobbra 135 tollatle
vége
```

```
eljárás ékkő :szín
balra 45 gyöngy :szín előre 30 jobbra 90 előre 30 balra 90 gyöngy :szín
hátra 30 jobbra 90 hátra 30 balra 45
vége
```

A díszített lánc kirajzolásához a nehézséget a váltakozó színek meghatározása jelenti. Figyelembe kell venni a lánc szemének páros vagy páratlan számát, illetve a középső és legszélső szemek közötti szemek párosságát vagy páratlanságát.

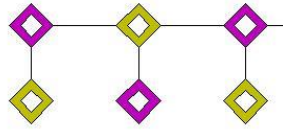
```
eljárás díszített :db
jobbra 45
gyöngy "szürke előre 30 jobbra 90 előre 30 balra 45 előre 60
ha :db>4 [váltakozó egészhányados :db-3 2 "igaz]
ha :db=3 [középső "sárga "szürke előre 60]
ha :db=4 [középső "sárga "szürke előre 60
középső "szürke "szürke előre 60]
ha maradék :db 2=1
[ha :db>4 [hakülönben maradék :db-3 2 =0
[középső "szürke "szürke előre 60
váltakozó egészhányados :db-3 2 "igaz]
[középső "sárga "szürke előre 60
váltakozó egészhányados :db-3 2 "hamis]]]
ha maradék :db 2=0
[ha :db>4 [hakülönben maradék :db-4 2=0
[középső "szürke "szürke előre 60
középső "sárga "szürke előre 60
váltakozó egészhányados :db-3 2 "hamis]
[középső "sárga "szürke e 60
középső "szürke "szürke előre 60
váltakozó egészhányados :db-3 2 "igaz]]]
balra 45 hakülönben maradék :db 2=0 [gyöngy "sárga][gyöngy "szürke]
tollatfel jobbra 45 hátra (:db-1)*60 hátra (:db-1)*30*gyök 2 balra 90
tollatle
vége
```

A középső ékköveket is tartalmazó elem.



```
eljárás középső :szín1 :szín2
balra 45 gyöngy :szín1 előre 30 jobbra 90 előre 30 jobbra 90
előre 30 balra 45 előre 30 ékkő :szín2 hátra 30 jobbra 45
hátra 30 balra 135
vége
```

A váltakozó a csüngő elemeket felváltva rajzolja ki, az egyik paraméter a kezdő szemet határozza meg.



```

eljárás váltakozó :db :egyik
  ha :db>0 [hakülönben :egyik [csüngő "sárga "lila]
           [csüngő "lila "sárga]
           előre 60 váltakozó :db-1 nem :egyik]
vége
  
```

A csüngő-t paraméterezzük a színekkel.

```

eljárás csüngő :szín1 :szín2
  balra 45 gyöngy :szín1 jobbra 90 előre 30 jobbra 45 előre 30
  balra 45 gyöngy :szín2 jobbra 45 hátra 30 balra 135 előre 30
  jobbra 45
vége
  
```

## Sorminta sorból és tükörképéből összeállítva

### Kígyó-sárkány

Egy pikkelyes kígyó szabályos háromszög alakú pikkelyeket hord a hátán. A kígyó hosszát az ívei számával (:ívdb) adjuk meg, s az első ív mindig felfelé kezdődik. Minden íven :db darab pikkely van. A sárkány hasonlít a kígyóra, csak neki mindkét oldalán vannak pikkelyek.

Készíts Logo eljárást kígyó (kígyó :ívdb :db :méret) és sárkány (sárkány :ívdb :db :méret) rajzolására!



kígyó 4 8 50



sárkány 3 8 50



sárkány 6 8 50

A sárkány eljárás a kígyó felhasználásával készül. Páratlan ívek esetén az utolsó ívet külön kell kezelni!

```

eljárás kígyó :ívdb :db :méret
  ismétlés egészhányados :ívdb 2 [jobb :db :méret bal :db :méret]
  ha maradék :ívdb 2 = 1 [jobb :db :méret]
vége
  
```

```

eljárás jobb :db :méret
  ismétlés :db [jobbra 90/:db ismétlés 3 [előre :méret balra 120]
              előre :méret jobbra 90/:db]
vége
  
```

```

eljárás bal :db :méret
  ismétlés :db [balra 90/:db ismétlés 3 [előre :méret balra 120]
              előre :méret balra 90/:db]
vége
  
```

```

eljárás sárkány :ívdb :db :méret
  kígyó :ívdb :db :méret balra 180
  hakülönben maradék :ívdb 2 = 0 [kígyó :ívdb :db :méret]
                                     [bal :db :méret kígyó :ívdb-1 :db :méret]
  jobbra 180
vége

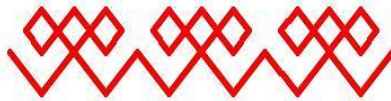
```

### Írásos hímzés, 2. verzió

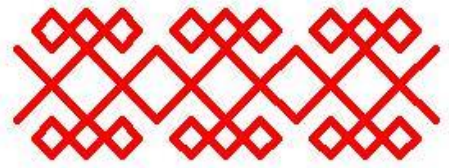
Készíts eljárást az ábrákon látható sorminták kirajzolására (sormintaA :db :méret, sormintaB :db :méret)! Írd meg az alapminta :méret kirajzolását is!



alapminta 50



sormintaA 3 50



sormintaB 3 50

A sormintaA kirajzolása nem okozhat gondot. A sormintaB-ben kétszer felhasználjuk a sormintaA-t. A második meghívásnál negatív értékkel használjuk a :méret paramétert, így megkapjuk a tükörképét.

```

eljárás minta :oldal
  előre 2*:oldal jobbra 90 előre 4*:oldal balra 90
  előre :oldal balra 90 előre :oldal
  balra 90 előre :oldal*2 jobbra 90 előre 2*:oldal balra 90
  előre :oldal balra 90 előre :oldal
  balra 90 előre :oldal*4 jobbra 90 előre 2*:oldal
  tollatfel hátra 2*:oldal balra 90 hátra 2*:oldal jobbra 90
  hátra 2*:oldal balra 90 hátra 2*:oldal tollatle
vége

```

```

eljárás sormintaA :db :méret
  ismétlés :db [minta :méret tollatfel előre 2*:méret jobbra 90
                előre 2*:méret balra 90 előre 2*:méret jobbra 90
                előre 2*:méret balra 90 tollatle]
  tollatfel
  ismétlés :db [jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret
                jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret]
  tollatle
vége

```

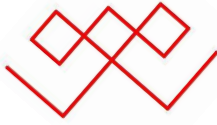
```

eljárás sormintaB :db :méret
  SormintaA :db :méret
  tollatfel jobbra 45 ismétlés :db [előre :méret*4*gyök 2] jobbra 45
  hátra 2*:méret jobbra 90 hátra 2*:méret balra 180 tollatle
  SormintaA :db -1*:méret
  tollatfel hátra 2*:méret jobbra 45 ismétlés :db [hátra :méret*4*gyök 2]
  balra 45 jobbra 90 előre 2*:méret balra 90 tollatle
vége

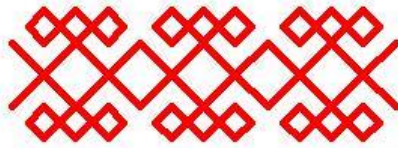
```

### Írásos hímzés, 3. verzió

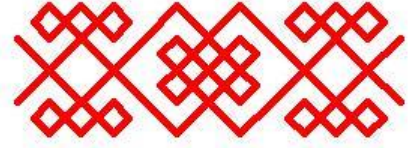
Készíts eljárást az ábrákon látható sorminták kirajzolására (sormintaA :db :méret, sormintaB :db :méret)! Írd meg az alapminta :méret kirajzolását is!



minta 50



sormintaA 3 50



sormintaB 3 50

A sormintaA elkészítéséhez először annak felét írjuk meg, ez a sorminta eljárás.

eljárás minta :oldal

```
előre 2*:oldal jobbra 90 előre 4*:oldal balra 90
előre :oldal balra 90 előre :oldal
balra 90 előre :oldal*2 jobbra 90 előre 2*:oldal balra 90
előre :oldal balra 90 előre :oldal
balra 90 előre :oldal*4 jobbra 90 előre 2*:oldal
tollatfel hátra 2*:oldal balra 90 hátra 2*:oldal jobbra 90
hátra 2*:oldal balra 90 hátra 2*:oldal tollatle
```

vége

eljárás sorminta :db :méret

```
ismétlés :db [minta :méret tollatfel előre 2*:méret jobbra 90
előre 2*:méret balra 90 előre 2*:méret jobbra 90
előre 2*:méret balra 90 tollatle]
```

tollatfel

```
ismétlés :db [jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret
jobbra 90 hátra 2*:méret balra 90 hátra 2*:méret]
```

tollatle

vége

eljárás sormintaA :db :méret

```
Sorminta :db :méret
tollatfel jobbra 45 ismétlés :db [előre :méret*4*gyök 2]
jobbra 45 hátra 2*:méret jobbra 90 hátra 2*:méret balra 180 tollatle
sorminta :db -1*:méret
tollatfel hátra 2*:méret jobbra 45 ismétlés :db [hátra :méret*4*gyök 2]
balra 45 jobbra 90 előre 2*:méret balra 90 tollatle
```

vége

A sormintaB-ben a minta hol balra, hol jobbra fordul. Ennek megfelelően el kell készíteni a minta eljárás forgatható változatát!

eljárás minta1 :oldal :fordul

```
előre 2*:oldal jobbra 90*:fordul előre 4*:oldal
balra 90*:fordul előre :oldal balra 90*:fordul előre :oldal
balra 90*:fordul előre :oldal*2 jobbra 90*:fordul
előre 2*:oldal balra 90*:fordul előre :oldal
balra 90*:fordul előre :oldal
balra 90*:fordul előre :oldal*4 jobbra 90*:fordul
előre 2*:oldal
tollatfel balra 45 hátra 4*:oldal*gyök 2 balra 45 tollatle
```

vége

A sormintaB elkészítéséhez szükség van egy hol jobbra, hol balra forduljon a minta felváltva.

eljárás sorminta2 :db :méret :fordul

```
ha :db>0 [minta :méret :fordul tollatfel jobbra 45
előre 4*:méret*gyök 2 tollatle jobbra 45
Sorminta2 :db-1 :méret-1*:fordul]
```

vége

A sormintaB-ben a páros és páratlan :db értéknél máshogy kell visszatérni a kiinduló helyzetbe!

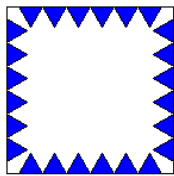
```

eljárás sormintaB :db :méret
  Sorminta2 :db :méret 1
  tollatfel
  hakülönben maradék :db 2 = 0
    [jobbra 45 hátra 4*:db*:méret*gyök 2 balra 45]
    [balra 45 hátra 4*:db*:méret*gyök 2 balra 45]
  előre 2*:méret jobbra 90 hátra 2*:méret
  tollatle sorminta2 :db :méret-1 tollatfel
  hakülönben maradék :db 2 = 1
    [jobbra 45 hátra 4*:db*:méret*gyök 2 balra 45]
    [balra 45 hátra 4*:db*:méret*gyök 2 balra 45 ]
  jobbra 90 előre 2*:méret balra 90 hátra 2*:méret tollatle
vége
  
```

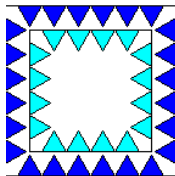
## Sorminta több sorból

### Négyzetek

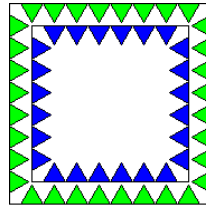
Készíts Logo eljárásokat „fogazott” négyzetek rajzolására! A színes fogakat a háromszög :hossz :szín eljárás rajzolja! A négyzet :db :hossz :szín eljárás olyan négyzetet rajzol, amelynek oldalain :db darab háromszög alakú, :hossz méretű :szín színű fog van. A fogak a négyzet sarkától :hossz/2 távolságra kezdődnek. A belül :db :hossz eljárás a négyzet belsejébe is rajzol egy pontosan 2-2 foggal kisebb négyzetet. A kívül :db :hossz eljárás a négyzet köré is rajzol egy pontosan 2-2 foggal nagyobb négyzetet. A kívülbelül :db :hossz eljárás pedig a négyzet belsejébe és köré is rajzol egy-egy újabb négyzetet. Mindegyik eljárás olyan legyen, hogy a különböző négyzeteken különböző színű fogak legyenek!



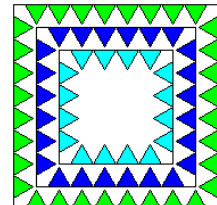
négyzet 6 40 "kék



belül 7 30



kívül 7 30



kívülbelül 7 30

Számoljuk ki, hogy mekkora a négyzet oldala, ha :db háromszöget helyezünk el az oldalán!

```

eljárás háromszög :h :szín
  ismétlés 3 [előre :h jobbra 120]
  tollatfel töltőszín! :szín
  jobbra 30 előre 3 tölt hátra 3 balra 30 tollatle
vége
  
```

```

eljárás négyzet :db :h :szín
  ismétlés 4 [előre :h/2
    ismétlés :db [háromszög :h :szín előre :h]
    előre :h/2 jobbra 90]
vége
  
```

```

eljárás belül :db :hossz
  négyzet :db :hossz "kék
  tollatfel jobbra 45 előre :hossz*gyök 2 balra 45 tollatle
  négyzet :db-2 :hossz "ibolya
  tollatfel jobbra 45 hátra :hossz*gyök 2 balra 45 tollatle
vége
  
```

```

eljárás kívül :db :hossz
  négyzet :db :hossz "kék
  tollatfel jobbra 45 hátra :hossz*gyök 2 balra 45 tollatle
  négyzet :db+2 :hossz "zöld
  tollatfel jobbra 45 előre :hossz*gyök 2 balra 45 tollatle
vége

```

Vegyük észre, hogy a kívülbelül eljáráshoz egyszerűen meg kell hívni a már elkészített belül és kívül eljárásokat!

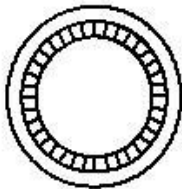
```

eljárás kívülbelül :db :hossz
  belül :db :hossz
  kívül :db :hossz
vége

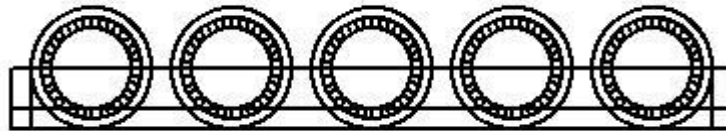
```

### Tányértartó

Készítsd el a megadott fali tányértartónak (tányértartó :db :méret) a rajzát a példa alapján, ahol a :db a tányértartóra helyezhető tányérok számát, a :méret pedig egy tányér méretét jelenti! A tányérok között hagyj ki arányosan helyet, ahogy a rajzon látod!



tányér 50



tányértartó 5 50

A tányértartó készítésénél vegyük észre, hogy a tányérok nem szabad érintkeznie egymással, így köztük hagyni kell mindenütt egy kis helyet! A két sor egyike a tányérok sora, a másik pedig a tányértartó rúd!

```

eljárás tányér :méret
  ív 360 :méret
  tollatfel jobbra 90 előre :méret/6 balra 90 tollatle
  ív 360 :méret*5/6 tollatfel jobbra 90 előre :méret/6 balra 90
  tollatle ív 360 :méret*2/3
  tollatfel jobbra 90 hátra :méret/6 balra 90 tollatle
  ismétlés 36 [ív 10 :méret/6*5 jobbra 90 előre :méret/6
    hátra :méret/6 balra 90]
  tollatfel jobbra 90 hátra :méret/6 balra 90 tollatle
vége

```

```

eljárás ív :szög :r
  ismétlés :szög [előre 2*3,14156*:r/360 jobbra 1]
vége

```

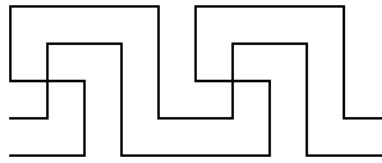
```

eljárás tányértartó :db :méret
  ismétlés 2 [előre 10 jobbra 90 e :db*( :méret*2+10)+10 jobbra 90]
  ismétlés 2 [előre :méret jobbra 90 előre 10 jobbra 90]
  előre :méret
  tollatfel jobbra 90 előre 10 balra 90 tollatle
  ismétlés :db [tányér :méret jobbra 90 előre 10+:méret*2 balra 90]
  jobbra 180
  ismétlés 2 [előre :méret jobbra 90 előre 10 jobbra 90]
  tollatfel előre 30 jobbra 90 előre :db*( :méret*2+10)+10 jobbra 90
  tollatle
vége

```

## Fríz

Egy régi épület falát szép mintacsík (fríz) díszíti. Készíts fríz :h :n eljárást, amely ilyen mintacsíkot rajzol!



fríz 30 1



fríz 10 3

Vegyük észre, hogy a minta két különböző L alakokból felépülő vonalból áll!

```

eljárás fríz :h :n
  jobbra 90
  absor :n :h tollatfel balra 90 előre :h/2 jobbra 90 tollatle
  basor :n :h tollatfel balra 90 hátra :h/2 tollatle
vége
  
```

```

eljárás absor :n :h
  ismétlés :n [A :h Bé :h] tollatfel hátra :n*5*:h tollatle
vége
  
```

```

eljárás basor :n :h
  ismétlés :n [Bé :h A :h] tollatfel hátra :n*5*:h tollatle
vége
  
```

```

eljárás A :hossz
  ismétlés 2 [előre :hossz balra 90]
  ismétlés 2 [előre :hossz jobbra 90] előre 2*:hossz jobbra 90
  előre 3*:hossz/2 balra 90 előre :hossz/2
vége
  
```

```

eljárás Bé :hossz
  előre :hossz/2 balra 90 ismétlés 2 [előre :hossz jobbra 90]
  előre 3*:hossz/2 balra 90 előre :hossz
vége
  
```

## Szálbehúzásos hímzés

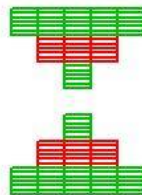
Készítsd el a négyzet :oldal :szín, a rész minta :oldal és a minta :oldal eljárásokat, majd ezeket felhasználva a következő sormintát rajzoló sorminta :db :oldal eljárást, ahol az :oldal a méretet, a :db pedig a sormintában szereplő elemek számát, határozza meg!



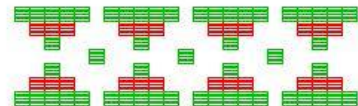
négyzet 10 2



rész minta 10



minta 10



sorminta 4 10

```

eljárás négyzet :méret :szín
  tollszín! :szín
  ismétlés 4 [előre :méret jobbra 90]
  ismétlés 5 [előre :méret/6 jobbra 90 előre :méret
              hátra :méret balra 90]
  hátra :méret/6*5
vége

```

```

eljárás részminta :méret
  négyzet :méret 2
  tollatfel előre :méret balra 90 előre :méret jobbra 90 tollatle
  ismétlés 3 [négyzet :méret 12 jobbra 90 előre :méret balra 90]
  tollatfel jobbra 90 hátra :méret*4 balra 90 előre :méret tollatle
  ismétlés 5 [négyzet :méret 2 jobbra 90 előre :méret balra 90]
  tollatfel jobbra 90 hátra :méret*3 balra 90 hátra :méret*2
  tollatle
vége

```

```

eljárás minta :méret
  részminta :méret
  tollatfel jobbra 180 előre :méret balra 90 előre :méret
  jobbra 90 tollatle
  részminta :méret
  tollatfel jobbra 180 előre :méret balra 90 előre :méret
  jobbra 90 tollatle
vége

```

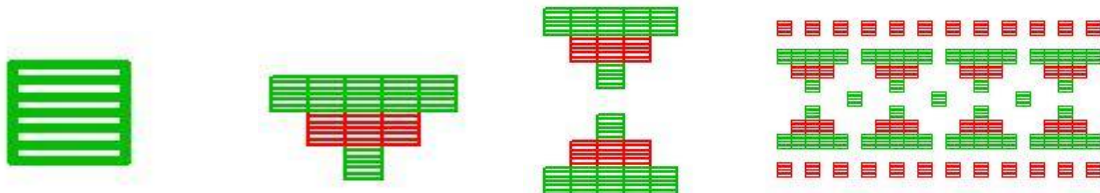
```

eljárás sorminta :db :méret
  ismétlés :db [minta :méret tollatfel jobbra 90
                előre :méret*6 balra 90 tollatle]
  tollatfel jobbra 90 hátra :db *:méret*6 tollatle
  ismétlés :db-1 [tollatfel előre :méret*3 tollatle
                  négyzet :méret 2 tollatfel
                  előre :méret*3 tollatle]
  tollatfel hátra :méret*(:db-1)*6 balra 90 tollatle
vége

```

### Szálbehúzásos hímzés, 2. verzió

Készítsd el a négyzet :oldal :szín :csíkdb, a részminta :oldal :csíkdb és a minta :oldal :csíkdb eljárásokat, majd ezeket felhasználva a következő sormintát rajzoló sorminta :db :oldal :csíkdb eljárást, ahol az :oldal a méretet, a :csíkdb a négyzetben megrajzolt vonalak számát, a :db pedig a sormintában szereplő elemek számát, határozza meg!



négyzet 10 2  
5

részminta 10 5

minta 10 5

sorminta 4 10 5

```

eljárás négyzet :méret :szín :csíkdarab
  tollszín! :szín
  ismétlés 4 [előre :méret jobbra 90]
  ismétlés :csíkdarab [előre :méret/(:csíkdarab+1) jobbra 90
                       előre :méret hátra :méret balra 90]
  hátra :méret/(:csíkdarab+1)*:csíkdarab
vége

```



```
eljárás rész minta :méret :csíkdarab
négyzet :méret 2 :csíkdarab
tollatfel előre :méret balra 90 előre :méret jobbra 90 tollatle
piros3 :méret :csíkdarab
tollatfel balra 90 előre :méret tollatle
zöld5 :méret :csíkdarab
tollatfel balra 90 hátra :méret*2 tollatle
vége
```

```
eljárás piros3 :méret :csíkdarab
ismétlés 3 [négyzet :méret 12 :csíkdarab jobbra 90
           előre :méret balra 90]
tollatfel jobbra 90 hátra :méret*4 tollatle
vége
```

```
eljárás zöld5 :méret :csíkdarab
ismétlés 5 [négyzet :méret 2 :csíkdarab jobbra 90
           előre :méret balra 90]
tollatfel jobbra 90 hátra :méret*3
vége
```

A minta két egymásnak fordított részmintából áll.

```
eljárás minta :méret :csíkdarab
ismétlés 2 [rész minta :méret :csíkdarab
           tollatfel jobbra 180 előre :méret balra 90 előre :méret
           jobbra 90 tollatle]
vége
```

A sorminta elkészítéséhez felhasználjuk a már megírt minta eljárást és a négyzetekből összeállítjuk a szélcsíkot és a középső dízsort.

```
eljárás sorminta :db :méret :csíkdarab
sor :db :méret :csíkdarab
középső :db :méret :csíkdarab
tollatfel előre 5*:méret jobbra 90 hátra 2*:méret tollatle
pirosak :db :méret :csíkdarab
tollatfel balra 90 hátra 11*:méret balra 90 előre :méret tollatle
pirosak :db :méret :csíkdarab
tollatfel hátra 3*:méret jobbra 90 előre 6*:méret tollatle
vége
```

```
eljárás sor :db :méret :csíkdarab
ismétlés :db [minta :méret :csíkdarab tollatfel jobbra 90
             előre :méret*6 balra 90 tollatle]
tollatfel jobbra 90 hátra :db*:méret*6 tollatle
vége
```

```
eljárás középső :db :méret :csíkdarab
ismétlés :db-1 [tollatfel előre :méret*3 tollatle
               négyzet :méret 2 :csíkdarab tollatfel
               előre :méret*3 tollatle]
tollatfel hátra :méret* (:db-1)*6 balra 90 tollatle
vége
```

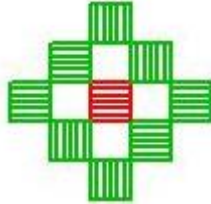
```
eljárás pirosak :db :méret :csíkdarab
ismétlés :db*3 [négyzet :méret 12 :csíkdarab tollatfel
               előre :méret*2 tollatle]
vége
```

### Szálbehúzásos hímzés, 3. verzió

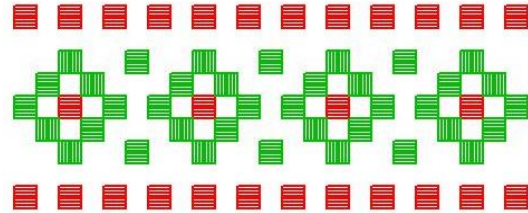
Készítsd el a négyzet :szín :oldal :csíkdb és a minta :oldal :csíkdb eljárásokat, majd ezeket felhasználva a sormintát rajzoló sorminta :db :oldal :csíkdb eljárást, ahol az :szín az alapelem színét, az :oldal a méretet, a :csíkdb a négyzetben meg-rajzolt vonalak számát, a :db pedig a sormintában szereplő elemek számát, határozza meg! A színeket tetszőlegesen választhatod meg!



négyzet 2 50 5



minta 50 5



sorminta 4 50 5

```
eljárás négyzet :méret :szín :csíkdarab
tollszín! :szín
ismétlés 4 [előre :méret jobbra 90]
ismétlés :csíkdarab [előre :méret/(:csíkdarab+1) jobbra 90
                    előre :méret hátra :méret balra 90]
hátra :méret/(:csíkdarab+1)*:csíkdarab
vége
```

A mintát a négyzetekből építjük fel.

```
eljárás minta :méret :csíkdarab
négyzet :méret 12 :csíkdarab
tollatfel jobbra 90 hátra 2*:méret balra 90 tollatle
ismétlés 3 [négyzet :méret 2 :csíkdarab tollatfel előre :méret
            jobbra 90 előre :méret balra 90 tollatle]
tollatfel hátra :méret*2 tollatle
négyzet :méret 2 :csíkdarab
tollatfel jobbra 90 előre :méret balra 90 hátra :méret
tollatle négyzet :méret 2 :csíkdarab
tollatfel jobbra 90 hátra :méret balra 90 hátra :méret tollatle
négyzet :méret 2 :csíkdarab
tollatfel hátra :méret jobbra 90 hátra :méret balra 90 tollatle
négyzet :méret 2 :csíkdarab
tollatfel előre :méret balra 90 előre :méret jobbra 90 tollatle
négyzet :méret 2 :csíkdarab
tollatfel előre :méret jobbra 90 előre :méret balra 90 tollatle
vége
```

```

eljárás sorminta :db :méret :csíkdarab
ismétlés :db [minta :méret :csíkdarab tollatfel jobbra 90
             előre :méret*6 balra 90 tollatle]
tollatfel jobbra 90 hátra :db*:méret*6 balra 90
előre 2*:méret tollatle
ismétlés :db-1 [tollatfel jobbra 90 előre :méret*3 balra 90
                tollatle négyzet :méret 2 :csíkdarab
                tollatfel jobbra 90 előre :méret*3 balra 90
                tollatle]
tollatfel hátra :méret*4 tollatle
ismétlés :db-1 [tollatfel balra 90 előre :méret*3 jobbra 90
                tollatle négyzet :méret 2 :csíkdarab
                tollatfel balra 90 előre :méret*3 jobbra 90
                tollatle]
tollatfel előre :méret*2
hátra 4*:méret jobbra 90 hátra 2*:méret balra 90 tollatle
ismétlés :db*3 [négyzet :méret 12 :csíkdarab tollatfel
                jobbra 90 előre :méret*2 balra 90 tollatle]
tollatfel jobbra 90 hátra 2*:méret balra 90 előre 8*:méret
tollatle
ismétlés :db*3 [négyzet :méret 12 :csíkdarab tollatfel
                balra 90 előre :méret*2 jobbra 90 tollatle]
tollatfel hátra 4*:méret jobbra 90 előre 4*:méret balra 90 tollatle
vége

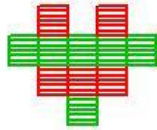
```

#### Szálbehúzásos hímzés, 4. verzió

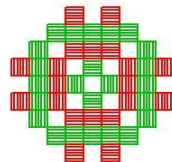
Készítsd el a négyzet :oldal :szín :csíkd, a részminta :oldal :csíkd, és a minta :oldal :csíkd eljárásokat is, majd ezeket felhasználva a következő sormintát rajzoló sorminta :db :oldal :csíkd eljárást, az :oldal a méretet, a :csíkd a négyzetben megrajzolt vonalak számát, a :db pedig a sormintában szereplő elemek számát határozza meg! A színeket tetszőlegesen választhatod meg!



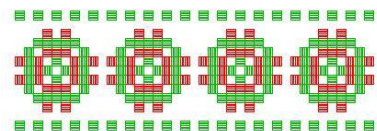
négyzet 20 2 5



részminta 10  
5



minta 10 5



sorminta 4 10 5

A részmintát négyzetekből építjük fel.

```

eljárás négyzet :méret :szín :csíkdarab
tollszín! :szín
ismétlés 4 [előre :méret jobbra 90]
ismétlés :csíkdarab [előre :méret/(:csíkdarab+1) jobbra 90
                    előre :méret hátra :méret balra 90]
hátra :méret/(:csíkdarab+1)*:csíkdarab
vége

```

```
eljárás rész minta :méret :csíkdarab
négyzet :méret 2 :csíkdarab
tollatfel előre :méret balra 90 előre :méret jobbra 90
tollatle
ismétlés 3 [négyzet :méret 12 :csíkdarab jobbra 90
             előre :méret balra 90]
tollatfel jobbra 90 hátra :méret*4 balra 90 előre :méret
tollatle
ismétlés 5 [négyzet :méret 2 :csíkdarab jobbra 90
             előre :méret balra 90]
tollatfel jobbra 90 hátra :méret *4 balra 90 előre :méret
tollatle
négyzet :méret 12 :csíkdarab
tollatfel jobbra 90 előre :méret*2 balra 90 tollatle
négyzet :méret 12 :csíkdarab
tollatfel hátra 3*:méret jobbra 90 hátra :méret balra 90
tollatle
vége
```

Megforgatjuk a szívet, így kapjuk meg a mintát!

```
eljárás minta :méret :csíkdarab
ismétlés 4 [rész minta :méret :csíkdarab tollatfel jobbra 90
            előre :méret tollatle]
vége
```

A sormintában a mintát használjuk fel és készítünk még hozzá két szélsort a négyzetekből!

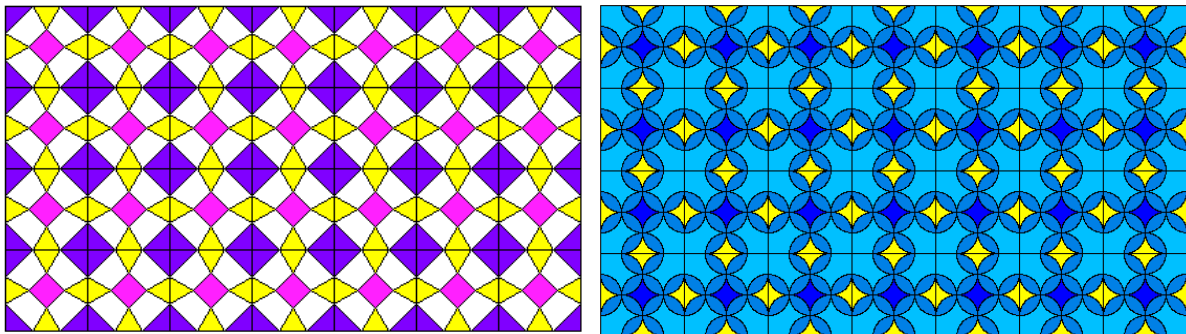
```
eljárás sorminta :db :méret :csíkdarab
ismétlés :db [minta :méret :csíkdarab tollatfel jobbra 90
              előre :méret*10 balra 90 tollatle]
tollatfel jobbra 90 hátra :db *:méret *10 balra 90
hátra 7*:méret jobbra 90 hátra 4*:méret balra 90 tollatle
ismétlés :db*5 [négyzet :méret 2 :csíkdarab tollatfel
                jobbra 90 előre :méret*2 balra 90 tollatle]
tollatfel jobbra 90 hátra 2*:méret balra 90 előre 12*:méret
tollatle
ismétlés :db*5 [négyzet :méret 2 :csíkdarab tollatfel
                balra 90 előre :méret*2 jobbra 90 tollatle]
tollatfel hátra 5*:méret jobbra 90 előre 6*:méret balra 90 tollatle
vége
```

## Mozaik – sorminták egymás fölé

A **mozaik** olyan művészeti technika és annak eredménye, amelynél kicsiny méretű színes üveg-, kő- vagy kavicsdarabokból állítják össze a képet vagy mintázatot (néha más anyagokat is használnak). A mozaikdarabokat cementtel, gipsszel rögzítik, esetleg a még nedves vakolatba nyomják bele.<sup>4</sup>

A Logo programozásban olyan mozaikokkal foglalkozunk, amelyek valamely geometrikus mintákból adott szabályszerűséggel épülnek fel, töltenek ki szabályos alakú területet.

Sokszor találkozunk ilyenekkel padló (parketta) burkolatoknál, térköveknél, csempéknél, üvegből készült vagy fába vésett mozaikoknál.



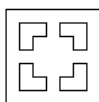
A mozaikok egy részében sormintákat helyezünk egymás fölé. Ilyen mozaik alkotásról olvashatunk az alábbi címen: [http://qtp.hu/mozaik/geometrikus\\_mintak\\_szerkesztese.php](http://qtp.hu/mozaik/geometrikus_mintak_szerkesztese.php).

### Négyzetmozaikok azonos alapelemekből

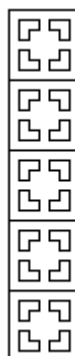
A legegyszerűbb esetben négyzet alakú elemekkel töltünk ki egy téglalapot, amikor a téglalap oldalainak hossza négyzet oldalainak egész számszorosa.

#### Csempe

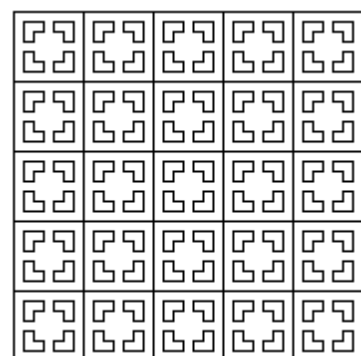
Az ábrán látható csempével szeretnénk egy falat kicsempézni. Készíts Logo eljárást a csempe rajzolására (csempe :méret, ahol :méret a négyzet alakú csempe oldalhossza), valamint hosszú és négyzetes falak csempézésére (hfal :db :méret, nfall :db :méret)!



csempe 50



hfal 5 35



nfall 5 35

<sup>4</sup> A Wikipédiából, a szabad enciklopédiából

A csempe egy nagyobb négyzet és egy kisebb négyzet sarkain levő négy szabályos alakzat:

```

eljárás csempe :méret
  ismétlés 4 [előre :méret*7 jobbra 90 tollatfel előre :méret
              jobbra 90 előre :méret balra 90 tollatle
              előre :méret*2 jobbra 90 előre :méret jobbra 90
              előre :méret balra 90 előre :méret jobbra 90
              előre :méret jobbra 90 előre :méret*2 tollatfel
              előre :méret jobbra 90 hátra :méret tollatle]
vége
  
```

A hfal egy sorminta, csak a szokásossal szemben függőleges irányban rajzolva:

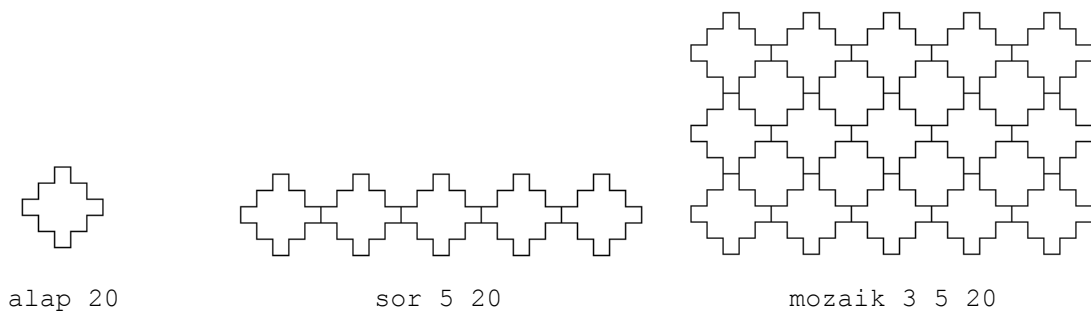
```

eljárás hfal :db :méret
  ismétlés :db [csempe :méret előre :méret*7]
  hátra :db*:méret*7
vége

eljárás nfal :db :méret
  ismétlés :db [hfal :db :méret jobbra 90 előre :méret*7 balra 90]
  jobbra 90 hátra :db*:méret*7 balra 90
vége
  
```

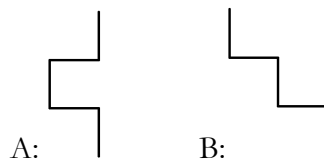
### Csempe kihagyott területtel

Készíts Logo programot, amely az ábrán látható alapelemet tudja rajzolni (alap :méret), az alapelemeket egymás mellé rakja egy sorba (sor :m :méret), illetve egymás mellé és fölé rakva egy területmintát alakít ki (mozaik :n :m :méret)!



Megjegyzés: a mozaik képen látszólag megjelenik egy, az alapelemtől különböző alakzat is, ezt azonban nem rajzoljuk, ez az alapelemek között kimaradt terület.

Az alapelem egy négyzet, aminek töröttvonalak az oldalai. Ez kétféleképpen is elképzelhető:



Ebben a megoldásban az elsőt választjuk:

```

eljárás alap :méret
  ismétlés 4 [előre :méret balra 90 előre :méret jobbra 90
              előre :méret jobbra 90 előre :méret balra 90
              előre :méret jobbra 90]
vége
  
```

```

eljárás sor :m :méret
  ismétlés :m [alap :méret tollatfel jobbra 90 előre :méret*5
                balra 90 tollatle]
  tollatfel jobbra 90 hátra :méret*5*:m balra 90 tollatle
vége
  
```

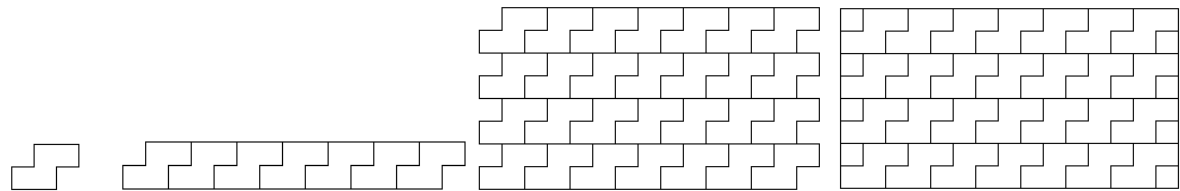
```

eljárás mozaik :n :m :méret
  ismétlés :n [sor :m :méret tollatfel előre :méret*5 tollatle]
  tollatfel hátra :méret*5*:n tollatle
vége
  
```

### Csempe külső szegély felhasználásával

Készíts Logo eljárást, amely az itt megadott téglából falat tud építeni! A téglát :méret eljárás egyetlen téglát rajzoljon, ahol :méret a téglát legrövidebb vonalának hossza! A téglát alja és teteje pedig  $2 * :méret$  hosszúságú legyen!

A sor :n :méret eljárás :n darab téglát rajzoljon egymás mellé, a mozaik :m :n :méret pedig :m sorból álló falat! A fal :m :n :méret egy téglalapba foglalt mozaik legyen!



csempe 20                      sor 7 20                      mozaik 4 7 20                      fal 4 7 20

```

eljárás téglát :méret
  ismétlés 2 [előre :méret jobbra 90 előre :méret balra 90
              előre :méret jobbra 90 előre 2*:méret jobbra 90]
vége
  
```

```

eljárás sor :n :méret
  ismétlés :n [téglát :méret jobbra 90 előre 2*:méret balra 90]
  tollatfel jobbra 90 hátra :n*2*:méret balra 90 tollatle
vége
  
```

```

eljárás mozaik :m :n :méret
  ismétlés :m [sor :n :méret előre 2*:méret]
  tollatfel hátra :m*2*:méret tollatle
vége
  
```

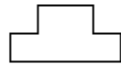
A mozaik nem fedi le a kiinduló téglalapot, ezért azt külön rajzoljuk köré.

```

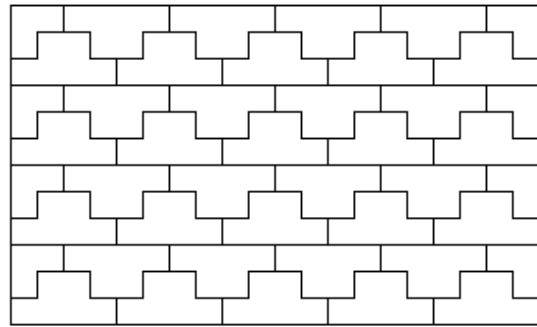
eljárás fal :m :n :méret
  mozaik :m :n :méret
  ismétlés 2 [előre 2*:m*:méret jobbra 90 előre (:n*2+1)*:méret jobbra 90]
vége
  
```

### Csempe látszólag szembefordított sorokkal

Készíts Logo eljárást, amely az itt megadott téglából falat tud építeni! A téglát :méret eljárás egyetlen téglát rajzoljon, ahol :méret a téglát legrövidebb vonalának hossza! A téglát alja  $4 * :méret$ , teteje pedig  $2 * :méret$  hosszú legyen! A fal :m :n :méret  $2 * :m$  sorban soronként :n téglából álló falat rajzoljon!

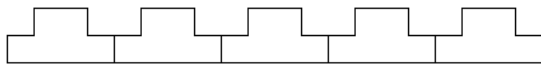


tégla 40

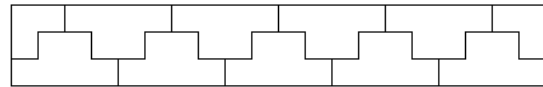


fal 4 5 40

A feladat felfogható úgy, hogy a páratlan sorszámú sorok normál állású téglákból vannak, a páros sorszámúak pedig eggyel kevesebb fordított állásúból, amit a végén egy-egy féltégla zár le.



sor 5 40



fal 1 5 40

eljárás tégla :méret

```
előre :méret jobbra 90 előre :méret balra 90 előre :méret
jobbra 90 előre 2*:méret jobbra 90 előre :méret balra 90
előre :méret jobbra 90 előre :méret jobbra 90 előre 4*:méret
jobbra 90
```

vége

eljárás sor :n :méret

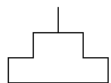
```
ismétlés :n [tégla :méret jobbra 90 előre 4*:méret balra 90]
vége
```

eljárás fal :m :n :méret

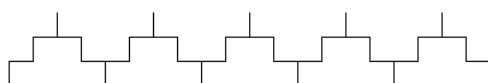
```
ismétlés :m [sor :n :méret előre 3*:méret balra 90
előre 2*:méret balra 90 sor :n-1 :méret jobbra 90
előre 2*:méret balra 90 előre 2*:méret jobbra 180
előre 2*:méret]
```

vége

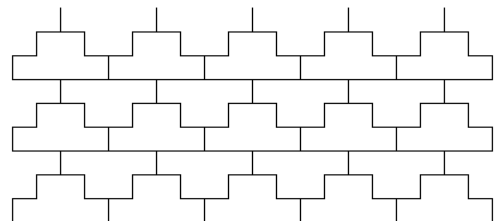
Sokkal egyszerűbb azonban a következő elképzelés:



csempe 40



sor 5 40



mozaik 3 5 40

Ezt már csak körbe kell venni egy téglalappal és készen is vagyunk!

eljárás tégla :méret

```
előre :méret jobbra 90 előre :méret balra 90 előre :méret jobbra 90
előre :méret balra 90 előre :méret hátra :méret jobbra 90
előre :méret jobbra 90 előre :méret balra 90 előre :méret jobbra 90
előre :méret jobbra 90 előre 4*:méret jobbra 90
```

vége



```

eljárás sor :n :méret
  ismétlés :n [tégla :méret jobbra 90 előre 4*:méret balra 90]
  tollatfel jobbra 90 hátra :n*4*:méret balra 90 tollatle
vége

eljárás mozaik :m :n :méret
  ismétlés :m [sor :n :méret tollatfel előre 3*:méret tollatle]
vége

eljárás fal :m :n :méret
  mozaik :m :n :méret
  ismétlés 2 [előre 3*:m*:méret jobbra 90 előre :n*4*:méret jobbra 90]
vége

```

### Asztal – két színnel festve, az egyik a háttér

Díszes, intarziás asztalt úgy készítünk, hogy egy téglalap alakú falapba illesztünk más fából készült betéteket. Egy lehetséges betét például nyolcszögletű, melyet egyenes vonalak és körívek határolnak. Ebből egymás mellé helyezhetünk  $m$  darabot, így kialakul egy sor. A sorokból egymás fölé helyezhetünk  $n$  darabot, így kialakul az asztal.

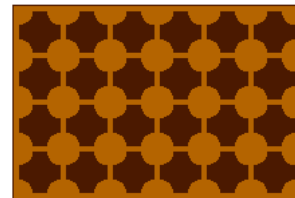
Készíts eljárásokat (alap, sor, asztal), amelyek az alábbi ábrákat rajzolják:



alap 20



sor 4 10



asztal 4 6 10

Az alapelemek  $3 \cdot h$  méretű négyzetlapból készülnek, negyedkörök levágásával. Színezéshez két-féle barna árnyalat RGB-kódját választottuk ki.

```

eljárás alap :h
  ismétlés 4 [előre :h jobbra 90 negyedkör :h jobbra 90]
  töltőszín! [75 25 0]
  tollatfel jobbra 45 előre :h tölt hátra :h balra 45 tollatle
vége

```

```

eljárás negyedkör :h
  ismétlés 90 [előre 3,14159*:h/180 balra 1]
vége

```

Két alapelem között  $h/2$  méretű rés van.

```

eljárás sor :m :h
  ismétlés :m [alap :h tollatfel jobbra 90 előre 3,5*:h balra 90
    tollatle]
  tollatfel jobbra 90 hátra :m*3,5*:h balra 90 tollatle
vége

```

Az asztal egymás fölé illesztett sorokból áll, amelyeket  $h/2$  távolságra egy világosabb barna téglalap vesz körül.

```

eljárás asztal :n :m :h
  ismétlés :n [sor :m :h tollatfel előre 3,5*:h tollatle]
  tollatfel hátra :n*3,5*:h+1,5*:h balra 90 előre :h/2 jobbra 90 tollatle
  ismétlés 2 [előre :n*3,5*:h+:h/2 jobbra 90
             előre :m*3,5*:h+:h/2 jobbra 90]
  töltőszín! [180 100 0]
  tollatfel jobbra 45 előre :h/2 tölt hátra :h/2 balra 45 tollatle
vége

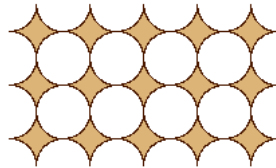
```

### Padló – két szín, újrafestéssel

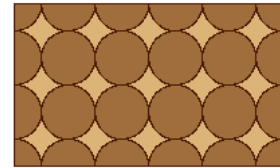
Egy régi épület padlóját mozaikminta díszíti. Készíts Logo eljárást (padló :m :n :h), amely ilyen mintát rajzol!



alap 50



mozaik 3 5 20



padló 3 5 20

A feladat megoldása hasonlíthatna az előzőre, van azonban egy lényeges különbség. Az előző feladatbeli asztalnál az alap-téglalapot egy menetben kiszínezhettük. Itt, ha a mozaikkal készen vagyunk, be kell járni a köröket és kitölteni őket – ez a félkörök miatt kicsit nehéz!

```

tanuld padló :m :n :h
  ismétlés 2 [előre :m*:h jobbra 90 előre :n*:h jobbra 90]
  tollatfel jobbra 90 előre :h/2 balra 90 tollatle
  mozaik :m :n :h kitölt :m :n :h
vége

tanuld mozaik :m :n :h
  ismétlés :m [sor :n :h tollatfel előre :h tollatle]
  tollatfel hátra :m*:h tollatle
vége

eljárás sor :m :h
  ismétlés :m [alap :h tollatfel jobbra 90 előre :h balra 90 tollatle]
  tollatfel tollatfel jobbra 90 hátra :m*:h balra 90 tollatle tollatle
vége

eljárás alap :h
  ismétlés 4 [negyedkör :h jobbra 180]
  töltőszín! [220 180 120] tollatfel előre :h/2 tölt hátra :h/2 tollatle
vége

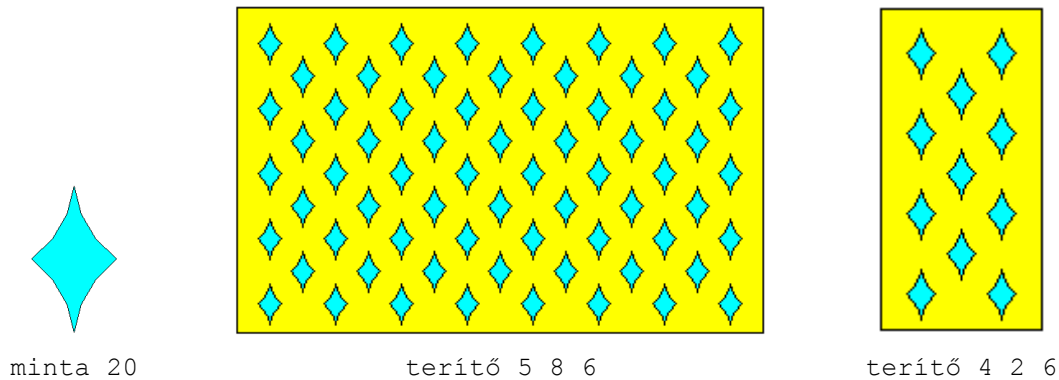
eljárás negyedkör :h
  ismétlés 90 [előre 3,14159*:h/360 balra 1]
vége

eljárás kitölt :m :n :h
  tollatfel töltőszín! [160 110 60]
  ismétlés :m [balra 45 előre :h/4 tölt hátra :h/4 jobbra 135
             ismétlés :n [balra 45 előre :h/4 tölt hátra :h/4 jobbra 45
                          előre :h]
             hátra :n*:h balra 90 előre :h]
  jobbra 45 hátra :h/4 tölt előre :h/4 jobbra 45
  ismétlés :n [jobbra 45 előre :h/4 tölt hátra :h/4 balra 45 előre :h]
  hátra :n*:h hátra :m*:h tollatle
vége

```

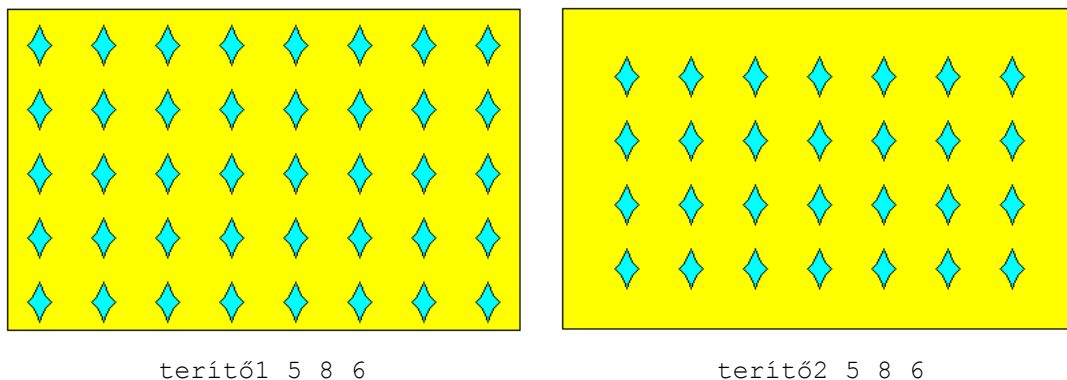
### Terítő – két mozaik egymáson

Egy sárga terítőn hosszúkás kék minták vannak. Az egyes mintaelemek hegyesebb végüknél 30, a tompábbnál 90 fokos szögűek. Az egyes oldalakon 2 törés van, ezek 165 fokos szöget zárnak be. Ha az egyenes darabok hosszát  $x$ -szel jelöljük, akkor a terítőn az egyes elemek egymástól  $8 \cdot x$  távolságra vannak soronként és oszloponként is, továbbá közöttük átlósan is található egy-egy elem. A terítőt egyszerű szegély keretezi szimmetrikusan. Készíts Logo eljárást (terítő :n :m :x), amely egy olyan terítőt rajzol, ahol egymás fölött :n elem, egymás mellett pedig :m elem található és köztük átlósan is vannak elemek!



A feladatban az tűnik bonyolultnak, hogy minden páros sorszámú sor eggyel kevesebb elemből áll és jobbra el van tolva az alatta levőhöz képest.

A megoldás egyszerűbb: képzeljünk el két mozaikot!



Tegyük egymásra a kettőt és már készen is vagyunk!

```

eljárás terítő :sdb :odb :méret
  töltőszín! 11
  téglalap 8*:sdb*:méret 8*:odb*:méret
  tollatfel jobbra 90 előre 4*:méret balra 90 előre :méret
  mozaik :sdb :odb :méret
  előre 4*:méret jobbra 90 előre 4*:méret balra 90
  mozaik :sdb-1 :odb-1 :méret
  jobbra 90 hátra 4*:méret balra 90 hátra 4*:méret hátra :méret/2
  tollatle töltőszín! 14 tölt tollatfel előre :méret/2
vége

eljárás téglalap :x :y
  ismétlés 2 [előre :x jobbra 90 előre :y jobbra 90]
vége
  
```

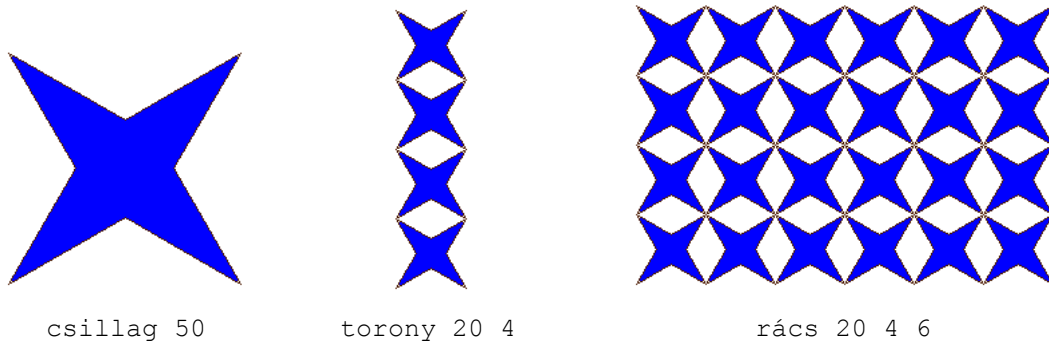
```
eljárás mozaik :sdb :odb :méret
  ismétlés :sdb [sor :odb :méret előre 8*:méret]
  hátra :sdb*:méret*8
vége

eljárás sor :db :méret
  ismétlés :db [minta :méret jobbra 90 előre 8*:méret balra 90]
  jobbra 90 hátra :db*8*:méret balra 90
vége

eljárás minta :méret
  tollatle
  ismétlés 2 [ismétlés 3 [balra 15 előre :méret] jobbra 90
              ismétlés 3 [előre :méret balra 15] jobbra 180]
  tollatfel előre :méret tölt hátra :méret
vége
```

### Csillagok

Csillagokból érdekes mintákat állíthatunk össze. Írj eljárást a csillag :h, a torony :h :s és a rács :h :s :o alakzatok megrajzolására! Legyen :h a 4-ágú csillag oldalainak hossza, a :s a torony és a rács magassága, a :o pedig a rács szélessége! A csillag hegyeinek szöge 30 fokos.



Itt a négyzet alapú mozaikban a négyzet oldalhosszát nehéz kiszámolni. Ha nincs rá szükségünk, akkor haladhatunk az alapelemek oldalain.

```
eljárás csillag :h
  jobbra 30
  ismétlés 4 [előre :h balra 60 előre :h jobbra 150]
  tollatfel jobbra 15 előre :h tölt hátra :h balra 45 tollatle
vége

eljárás torony :h :db
  ismétlés :db [csillag :h
                jobbra 30 előre :h balra 60 előre :h jobbra 30]
  ismétlés :db [balra 30 hátra :h jobbra 60 hátra :h balra 30]
vége

eljárás rács :h :s :o
  töltőszín! [0 0 255]
  ismétlés :o [torony :h :s
                jobbra 60 előre :h jobbra 60 előre :h balra 120]
  ismétlés :o [balra 60 előre :h balra 60 előre :h jobbra 120]
vége
```

### Mozaik kétféle sorból

#### Csempe

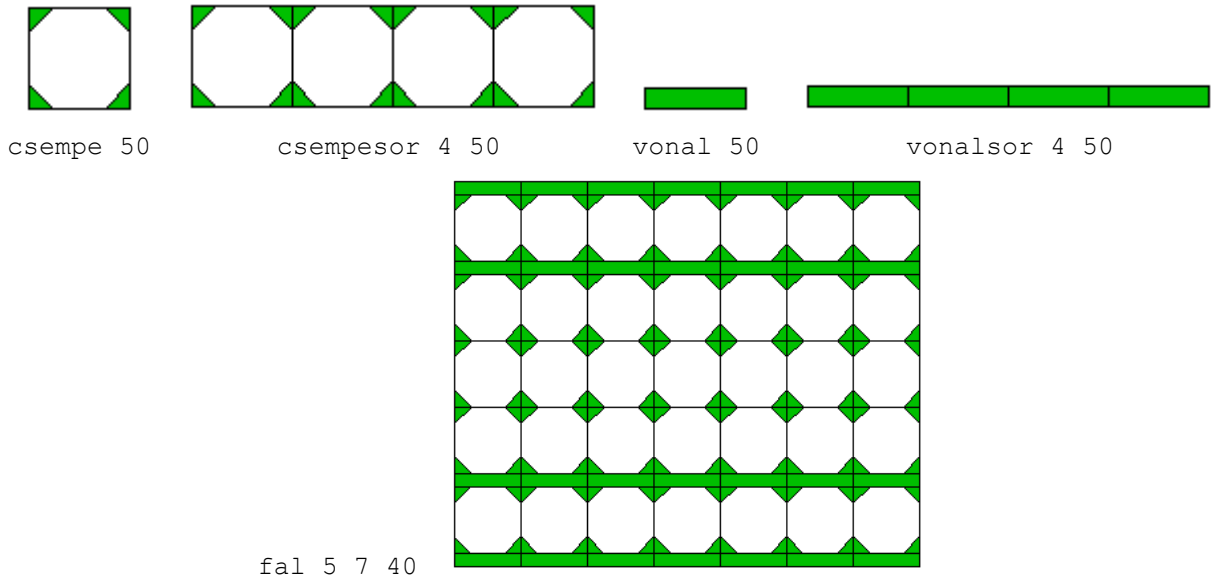
Egy fürdőszoba falat a mellékelt mintázatú csempével szeretnénk befedni. Készíts Logo eljárásokat a fal és az egyes alapelemek rajzolására!

A csempe :hossz eljárás egyetlen csempét rajzoljon, amely egy :hossz oldalhosszúságú négyzet, melynek sarkaiban zöld derékszögű háromszögek vannak (befogójuk az oldalhossz negyede)!

A csempesor :db :hossz eljárás egy sort rajzol, :db darab :hossz méretű csempéből.

A vonal :hossz eljárás egy :hossz szélességű, negyedannyi magasságú zöld téglalapot rajzol, a vonalsor :db :hossz pedig egy :db darab :hossz méretű vonalból álló sort.

A fal :n :m :hossz eljárás :n csempesort rajzol, amely :m darab :hossz méretű csempéből áll. Az alsó és a felső csempesort mindkét oldaláról egy-egy zöld vonalsor határolja.



```
eljárás csempe :h
  ismétlés 4 [alap2 :h/4 előre :h jobbra 90]
vége
```

```
eljárás alap2 :h
  előre :h jobbra 135 előre :h*gyök 2 jobbra 135 előre :h
  jobbra 90 tollatfel töltőszín! 10 jobbra 45 előre 3 tölt
  hátra 3 balra 45 tollatle
vége
```

```
eljárás csempesor :csdb :hossz
  ismétlés :csdb [csempe :hossz jobbra 90 előre :hossz balra 90]
  jobbra 90 hátra :csdb*:hossz balra 90
vége
```

```
eljárás vonal :h
  ism 2 [előre :h/5 jobbra 90 előre :h jobbra 90]
  tollatfel töltőszín! 10
  jobbra 45 előre 3 tölt hátra 3 balra 45 tollatle
vége
```

```
eljárás vonalsor :csdb :hossz
  ismétlés :csdb [vonalsor :hossz jobbra 90 előre :hossz balra 90]
  jobbra 90 hátra :csdb*:hossz balra 90
vége
```

```

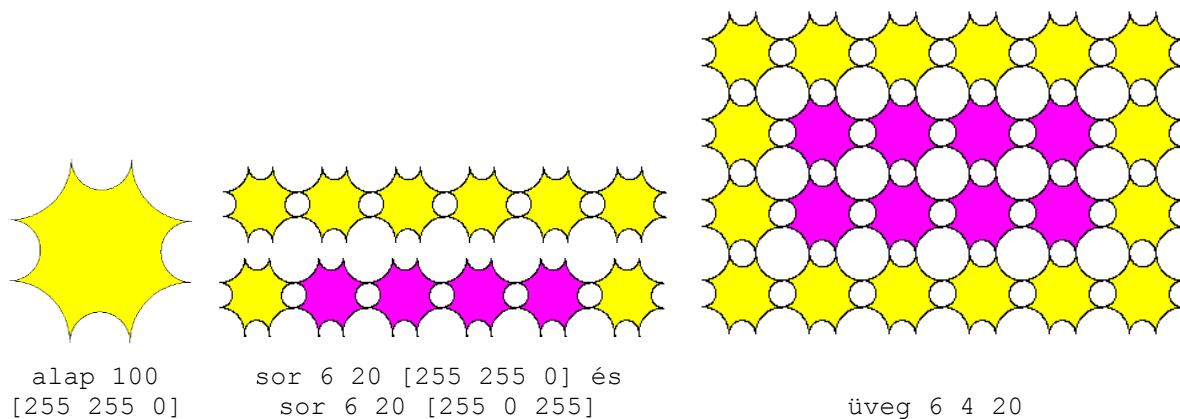
eljárás fal :n :m :hossz
  vonalsor :m :hossz előre :hossz/5
  csempesor :m :hossz előre :hossz
  vonalsor :m :hossz előre :hossz/5
  ismétlés :n-2 [csempesor :m :hossz előre :hossz]
  vonalsor :m :hossz előre :hossz/5
  csempesor :m :hossz előre :hossz
  vonalsor :m :hossz előre :hossz/5
  hátra :n*:hossz+4*:hossz/5
vége

```

### Ablak – a külső elemek más színűek

A középkorban díszes ablakokat csak nagyon kicsi üvegtáblákból tudtak kirakni. Egy lehetséges üvegtábla például nyolcszögletű, negyed- és félkörökkel határolt alakzat. Ebből egymás mellé helyezhetünk  $M$  darabot, így kialakul egy üvegtábla sor. A sorokból egymás fölé helyezhetünk  $N$  darabot, így kialakul az üvegtábla.

Készíts eljárásokat (alap, sor, üveg), amelyek az alábbi ábrákat rajzolják:



Az alapelemünk negyed- és félkörívекből áll, amit a megrajzolás után a paraméterként kapott színnel befestünk.

```

eljárás alap :h :szín
  töltőszín! :szín
  ismétlés 4 [jobbra 90 félkör :h/2 jobbra 180 negyedkör :h jobbra 90]
  tollatfel jobbra 90 előre :h tölt hátra :h balra 90 tollatle
vége

```

```

eljárás negyedkör :h
  ismétlés 90 [előre 3,14159*:h/180 balra 1]
vége

```

```

eljárás félkör :h
  ismétlés 180 [előre 3,14159*:h/180 balra 1]
vége

```

A feladathoz látszólag kétféle sor kell. De igazából egy is elég, ha a sor egy alapelemből, utána egy  $m-2$  hosszú sorból és még egy alapelemből áll. A két szélső elem színe legyen sárga, a többié pedig a paraméterként kapott szín!

```

eljárás sor :m :h :szín
  alap :h [255 255 0] eltol :h
  ismétlés :m-2 [alap :h :szín eltol :h]
  alap :h [255 255 0] eltol :h
  tollatfel balra 180 eltol :m*:h jobbra 180 tollatle
vége

```

```

eljárás eltol :h
  tollatfel jobbra 90 előre 3*:h balra 90 tollatle
vége
  
```

Az üveg megrajzolásakor az első és utolsó sorban közbülső színnek a sárgát adjuk meg, a többinél pedig a magentát.

```

eljárás üveg :n :m :h
  sor :m :h [255 255 0] tollatfel előre 3*:h tollatle
  ismétlés :n-2 [sor :m :h [255 0 255] tollatfel előre 3*:h tollatle]
  sor :m :h [255 255 0] tollatfel hátra (:n-1)*:h*3 tollatle
vége
  
```

## Mozaik kétféle elemből

Ha a mozaik kétféle elemből áll, akkor egy ciklus lépésben két elemet, illetve két sort rajzolunk. Páratlan lépésszám esetén az egyiket a cikluson kívül megismételjük még egyszer.

### Padló

Egy padló 30x30-as méretű csíkozott négyzetekből épül fel. Készítsd el a megrajzolásához szükséges Logo eljárásokat!

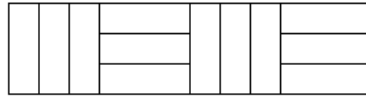
- függőleges :h a négyzetet függőlegesen harmadolja két vonal.  
vízszintes :h a négyzetet vízszintesen harmadolja két vonal.  
sorf :m :h egymás mellé elhelyezett :m darab négyzet, a páratlan sorszámúak függőleges, a páros sorszámúak pedig vízszintes csíkozásúak.  
sorv :m :h egymás mellé elhelyezett :m darab négyzet, a páratlan sorszámúak vízszintes, a páros sorszámúak pedig függőleges csíkozásúak.  
mozaik :n :m :h egymás fölé elhelyezett :n darab sor, a páratlan sorszámúak függőleges, a páros sorszámúak vízszintes csíkozású négyzettel kezdődnek.  
padló :n :m :h a mintának megfelelően egymás végébe elhelyezett 4 darab mozaik.



függőleges 20



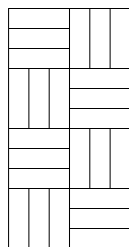
vízszintes 20



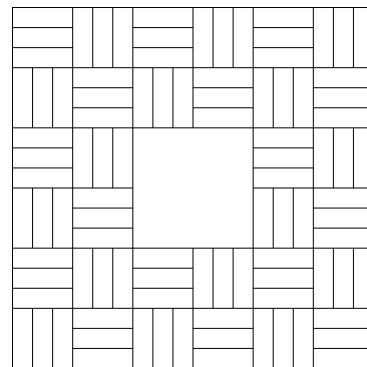
sorf 4 20



sorv 4 20



mozaik 4 2 10



padló 4 2 10

A kétféle alapelem függőleges, illetve vízszintes csíkozású négyzet.

```
eljárás függőleges :h
  ismétlés 4 [előre :h jobbra 90]
  ismétlés 2 [jobbra 90 előre :h/3 balra 90 előre :h hátra :h]
  jobbra 90 hátra 2*:h/3 balra 90
vége
```

```
eljárás vízszintes :h
  ismétlés 4 [előre :h jobbra 90]
  ismétlés 2 [előre :h/3 jobbra 90 előre :h hátra :h balra 90]
  hátra 2*:h/3
vége
```

Készítünk egy függőleges csíkozású elemmel kezdődő sort és egy vízszintes csíkozású elemmel kezdődő sort. Figyelni kell, hogy a sorok páros vagy páratlan számú elemből állnak-e!

```
eljárás sorf :n :h
  ismétlés :n/2 [függőleges :h jobbra 90 előre :h balra 90
                vízszintes :h jobbra 90 előre :h balra 90]
  ha l=maradék :n 2 [függőleges :h jobbra 90 előre :h balra 90]
  jobbra 90 hátra :n*:h balra 90
vége
```

```
eljárás sorv :n :h
  ismétlés :n/2 [vízszintes :h jobbra 90 előre :h balra 90
                függőleges :h jobbra 90 előre :h balra 90]
  ha l=maradék :n 2 [vízszintes :h jobbra 90 előre :h balra 90]
  jobbra 90 hátra :n*:h balra 90
vége
```

A mozaikban :n/2 darab sorf, illetve sorv szerepel. Ha :n páratlan, akkor még egy sorf-et kell rajzolni!

```
eljárás mozaik :n :m :h
  ismétlés :n/2 [sorf :m :h előre :h sorv :m :h előre :h]
  ha l=maradék :n 2 [sorf :m :h előre :h]
  hátra :n*:h
vége
```

A padló egy négyzet 4 oldalára elhelyezett mozaik.

```
eljárás padló :n :m :h
  ismétlés 4 [mozaik :n :m :h előre (:n+m)*:h jobbra 90]
vége
```

### Csillagok

Írj Logo eljárásokat a :h oldalhosszúságú, 20 fokos szögű 4-ágú festett csillag, valamint azokból készült alakzatok megrajzolására (csillag :h, csillagsor :h :o és minta :h :s :o, ahol :s a minta sorai, :o pedig az oszlopai száma)!

Ez is négyzet alapú mozaik lesz, de a négyzetek egymáshoz képest 45 fokkal el vannak forgatva, emiatt át is fedik egymást. De ezzel nem kell törődnünk, a négyzet helyére kerülő csillagok jól fognak érintkezni.





Kétféle elemet tartalmazó mozaik esetén az egyik lehetséges hozzáállás: kétféle sort definiálunk és felváltva hívjuk őket. A soron belül kétféle elemet – most csak a kétféle elem közötti elmozdulást – definiálunk és ezt is felváltva csináljuk.

```

eljárás minta :n :m :h :x :y :i
  ismétlés :n/2 [csillagsor :m :h xhely yhely irány
    ismétlés 2 [balra 10 előre :h balra 70 előre :h
      jobbra 170]
    jobbra 55 előre :h jobbra 170
    csillagsor2 :m :h xhely yhely irány
    balra 10 előre :h balra 70 előre :h jobbra 160 előre :h
    balra 125]
  ha 1=maradék :n 2 [csillagsor :m :h xhely yhely irány]
  tollatfel xyhely! :x :y irány! :i tollatle
vége
  
```

A csillagsor váltakozva egyenes állású és 45 fokban elforgatott csillagokból áll. A csillagsor2 hasonló, csak a kétféle állású csillag sorrendje más.

```

eljárás csillagsor :o :h :x :y :i
  ismétlés :o/2 [csillag :h jobbra 10 előre :h jobbra 70 előre :h
    balra 45 hátra :h jobbra 10 csillag :h tollatfel
    jobbra 10 előre :h jobbra 70 előre :h balra 160 előre :h
    jobbra 135 előre :h jobbra 70 előre :h balra 170
    tollatle]
  ha 1=maradék :o 2 [csillag :h]
  tollatfel xyhely! :x :y irány! :i tollatle
vége
  
```

```

tanuld csillagsor2 :o :h :x :y :i
  ismétlés :o/2 [csillag :h jobbra 10 előre :h jobbra 70 előre :h
    balra 160 előre :h jobbra 135 előre :h jobbra 70
    előre :h balra 170 csillag :h tollatfel jobbra 10
    előre :h jobbra 70 előre :h balra 45 hátra :h jobbra 10
    tollatle]
  ha 1=maradék :o 2 [csillag :h]
  tollatfel xyhely! :x :y irány! :i tollatle
vége
  
```

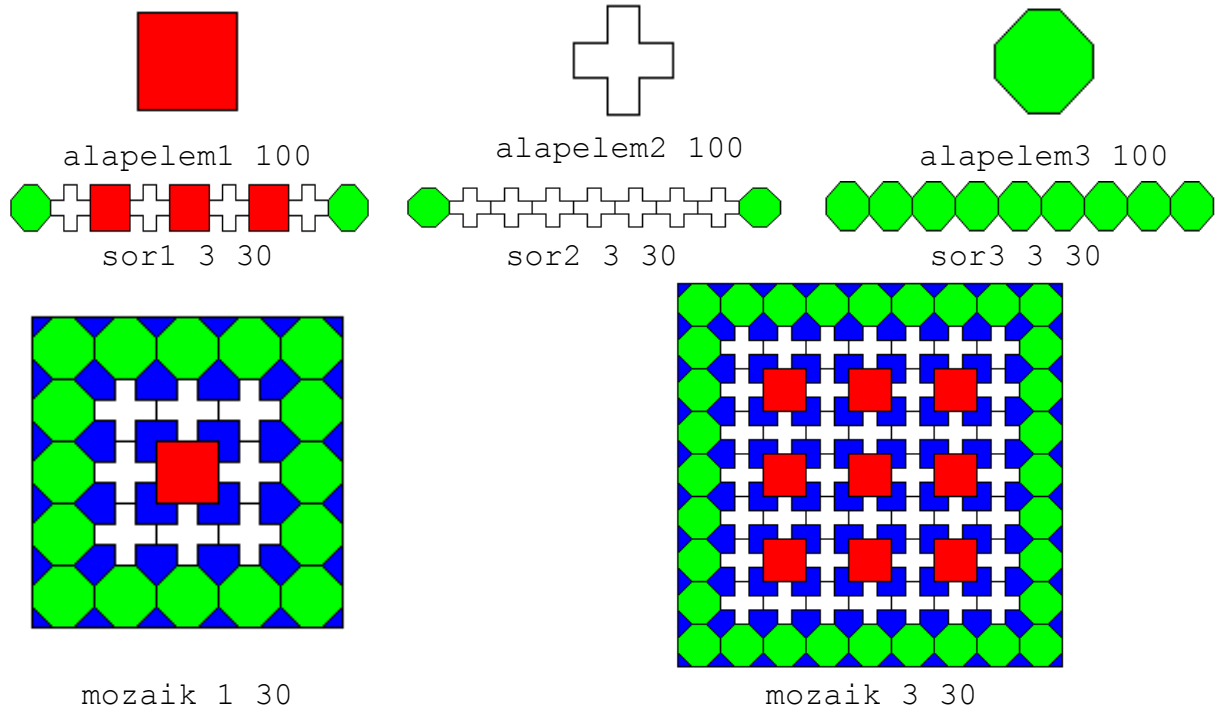
```

eljárás csillag csillag :h
  balra 10
  ismétlés 4 [előre :h balra 70 előre :h jobbra 160]
  tollatfel jobbra 10 előre :h tölt hátra :h tollatle
vége
  
```

### Mozaik – szegély is, kétféle elem is

Egy mozaikot háromféle alapelemből építünk fel (alapelem1 :h, alapelem2 :h, alapelem3 :h), ahol :h az alapelemek köré írható négyzet oldalának hossza. Az alapelemek sorokba

rendezhető (sor1 :m :h, sor2 :m :h, sor3 :m :h), ahol a sorok :m\*2+3 darab alapelemet tartalmaznak. A sorok felépítése az ábrán látható. Sorok alkalmas egymás mellé helyezésével készíts mozaikot (mozaik :m :h), amelynek belsejében :m\*2+3 sorban soronként :m piros négyzet található!



Ebben a mozaikban háromféle sort kell rajzolnunk! Ha a kék alaptéglalapot előre kirajzoljuk, akkor a festése sokkal egyszerűbb.

```

eljárás mozaik :n :a
  kékalap (:n*2+3)*:a
  sor3 :n :a tollatfel előre :a tollatle
  sor2 :n :a tollatfel előre :a tollatle
  ismétlés :n [sor1 :n :a tollatfel előre :a tollatle
               sor2 :n :a tollatfel előre :a tollatle]
  sor3 :n :a
vége

eljárás kékalap :h
  töltőszín! "kék"
  ismétlés 4 [előre :h jobbra 90]
  tollatfel jobbra 45 előre 5 tölt hátra 5 balra 45 tollatle
vége

eljárás sor1 :n :a
  alapelem3 :a eltol :a
  ismétlés :n [alapelem2 :a eltol :a alapelem1 :a eltol :a]
  alapelem2 :a eltol :a alapelem3 :a eltol :a
vége

eljárás eltol :a
  tollatfel jobbra 90 előre :a balra 90 tollatle
vége
    
```

```

eljárás sor2 :n :a
  alapelem3 :a eltol :a
  ismétlés :n*2+1 [alapelem2 :a eltol :a]
  alapelem3 :a tollatfel jobbra 90 hátra (:n*2+2)*:a balra 90
  tollatle
vége

eljárás sor3 :n :a
  ismétlés :n*2+3 [alapelem3 :a eltol :a]
  tollatfel jobbra 90 hátra (:n*2+3)*:a balra 90 tollatle
vége

eljárás alapelem1 :a
  ismétlés 4 [előre :a jobbra 90]
  tollatfel jobbra 45 előre 5 töltőszín! "piros tölt hátra 5
  balra 45 tollatle
vége

eljárás alapelem2 :a
  tollatfel előre :a/3 tollatle
  ismétlés 4 [előre :a/3 jobbra 90 előre :a/3 balra 90
             előre :a/3 jobbra 90]
  tollatfel jobbra 45 előre 5 töltőszín! "fehér tölt hátra 5
  balra 45 hátra :a/3 tollatle
vége

eljárás alapelem3 :a
  tollatfel előre :a/3 tollatle
  ismétlés 4 [előre :a/3 jobbra 45 előre :a/3*gyök 2 jobbra 45]
  tollatfel hátra :a/3 jobbra 45 előre :a/2 töltőszín! "zöld
  tölt hátra :a/2 balra 45 tollatle
vége

```

## Nem négyzet alapú mozaikok

Mozaik alapja nem csak négyzet lehet, az első példa azonban becsapós. Ha elforgatjuk a hálót 45 fokkal, akkor egy szabályos téglalapot lefedő, négyzet alapú mozaikot látunk.

A sík négyzeten kívül még kétféle szabályos sokszöggel fedhető le: egyenlő oldalú háromszöggel, illetve hatszöggel.

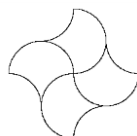
### Háló

Egy háló egy szeme speciális körívekből épül fel. A szemeket egy téglalap alakú területen fűzzük össze, :n sorba, soronként :m darab szemet.

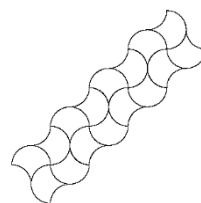
Készíts Logo eljárásokat (alap :sugár, szem :sugár, sor :m :sugár, háló :n :m :sugár) a feladat megoldására!



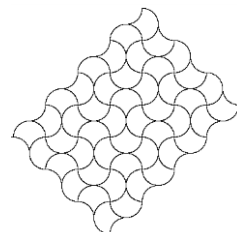
alap 40





szem 20



sor 4 10



háló 3 4 10

Az alap eljárás kirajzolja a  -t, a szem a háló egy szemét , a sor a háló egy sorát, a háló pedig a teljes mozaikot. A bal- és jobbívét az elemek kirajzolásánál, illetve a pozícionálásnál használjuk.

```
eljárás balív :fok :r
  ismétlés :fok*10 [előre :r*3,14159/180 balra .1]
vége
```

```
eljárás jobbív :fok :r
  ismétlés :fok*10 [előre :r*3,14159/180 jobbra .1]
vége
```

Az alapelem 3 körívből áll, megfelelően egymáshoz illesztve.

```
eljárás alap :r
  balív 90 :r jobbra 90 jobbív 180 :r jobbra 90 balív 90 :r jobbra 180
vége
```

A szem 4 alapelemből áll, egymáshoz képest eltolva és 90 fokkal elforgatva.

```
eljárás szem :r
  ismétlés 4 [alap :r balív 90 :r jobbra 90 jobbív 90 :r]
vége
```

```
eljárás sor :m :r
  ismétlés :m-1 [szem :r jobbív 90 :r balív 90 :r] szem :r
  jobbra 180 ismétlés :m-1 [jobbív 90 :r balív 90 :r] jobbra 180
vége
```

```
eljárás háló :n :m :r
  ismétlés :n-1 [sor :m :r balív 90 :r jobbív 90 :r] sor :m :r
  jobbra 180 ismétlés :n-1 [balív 90 :r jobbív 90 :r] jobbra 180
vége
```

## Hatszögmozaik

Készíts Logo eljárást, amely méhsejtekből különböző alakzatokat tud építeni! A hatszög :méret eljárás egyetlen méhsejtet rajzoljon, ahol :méret a hatszög oldalhossza!

A sor :n :méret eljárás :n darab méhsejtet rajzoljon egymás mellé!

A mozaik1 :n :méret, mozaik2 :n :méret eljárások pedig az alábbi ábrákat rajzolják, ahol :n az alsó sorban levő hatszögek száma, :méret pedig a hatszögek oldalhossza!



```
eljárás hatszög :hossz
  ismétlés 6 [előre :hossz jobbra 60]
vége
```

```

eljárás sor :n :hossz
  ismétlés :n [hatszög :hossz
                ismétlés 2 [balra 60 hátra :hossz] jobbra 120]
  ismétlés :n [balra 120 ismétlés 2 [előre :hossz jobbra 60]]
vége

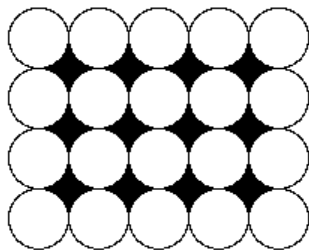
eljárás mozaik1 :n :hossz
  ismétlés :n [sor :n :hossz
                előre :hossz jobbra 60 előre :hossz balra 60]
vége

eljárás mozaik2 :n :hossz
  ismétlés :n/2 [sor :n :hossz előre :hossz jobbra 60 előre :hossz
                balra 60 sor :n :hossz tollatfel előre :hossz
                balra 60 előre :hossz jobbra 60 tollatle]
  ha 1=maradék :n 2 [sor :n :hossz]
vége

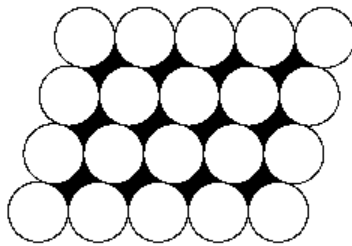
```

### Síkfedés körökkel

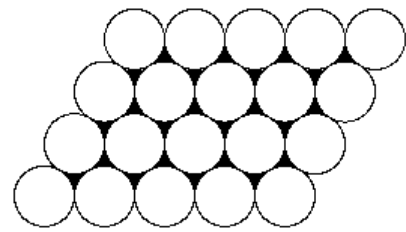
Egy mozaikot  $r$  sugarú körökből építünk fel. Megadjuk a sorok és oszlopok számát, valamint azt, hogy egy sor első köréhez képest hány fokkal van eltolva a következő (0 és 30 közötti szám). Az eltolásnál a sorokat, amennyire lehet, összecúsztatjuk. A körök közötti részeket befestjük. Készítsd el a mozaik  $m : n : r : \text{fok}$  eljárást, ami az alábbi ábrákat rajzolja! A  $\text{sor} : n : r$  eljárás pedig egy sort rajzol  $n$  darab  $r$  sugarú körből.



mozaik 4 5 20 0



mozaik 4 5 20 15



mozaik 4 5 20 30

```

eljárás körsor :n :r
  ismétlés :n [kör :r jobbra 90 előre 2*:r balra 90]
  jobbra 90 hátra 2*:n*r balra 90
vége

eljárás kör :r
  tollatle
  ismétlés 360 [előre :r*3,14159/180 jobbra 1]
  tollatfel
vége

```

Utólag elég nehéz a körök között kimaradt rések kitöltése, emiatt azt a megoldást választjuk (az első sor kivételével), hogy minden sor megrajzolása után kitöltjük az alatta levő sorközöket.

```

eljárás körmozaik :m :n :r :szög
  tollatfel körsor :n :r jobbra :szög előre 2*:r balra :szög
  ismétlés :m-1 [körsor :n :r sorokközött :n-1 :r jobbra :szög
                előre 2*:r balra :szög]
  jobbra :szög hátra 2*:m*r balra :szög tollatle
vége

eljárás körsor :n :r
  ismétlés :n [kör :r jobbra 90 előre 2*:r balra 90]
  jobbra 90 hátra 2*:n*r balra 90
vége

```

A sorok közötti „réseket töltjük ki”.

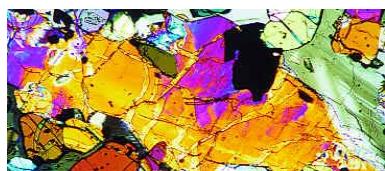
```
eljárás sorokközött :n :r
  ismétlés :n [körív :r 200 balra 90 előre 1 tölt hátra 1 jobbra 90
    körív :r 60 balra 90 előre 1 tölt hátra 1 jobbra 90
    körív :r 100 jobbra 90 előre 2*:r balra 90]
  jobbra 90 hátra 2*:n*:r balra 90
vége

eljárás körív :r :szög
  ismétlés :szög [előre :r*3,14159/180 jobbra 1]
vége
```

## Ásványok – molekulák – kristályok

A természetben sok szép ásványt, kristályt találhatunk. Mindegyiknek valamilyen jellemző szerkezete van. Ha az ásvány maga nem is, de a szerkezete mindenképpen rajzolható Logo programmal.

Nézzünk néhány példát, amelynek szerkezetét Logo-ban meg is rajzolhatjuk:



piroxén



csillám



amfibol



szilikát



béta kvarc



sassolin

### Sokszögrajzolásra épülő ásványok

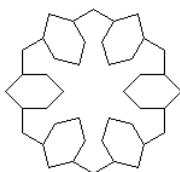
A legegyszerűbb esetekben sokszögeket kell rajzolnunk, ezekből esetleg sormintát, mozaikot készítenk.

#### Gyémántok

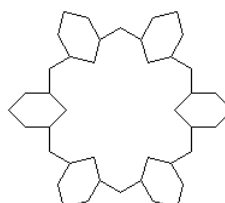
Készíts Logo programot, amely egy gyémántot, majd a mellékelt ábrának megfelelően kőtömbökbe zárt gyémántokat tud rajzolni! A kőtömb mindig szabályos hatszög alakú legyen, s az ábrákon látható összes szakasz egyforma hosszúságú! Négy eljárást készíts (gyémánt, tömb1, tömb2, tömb3 néven), melyeknek egyetlen paramétere a gyémánt oldalhossza!



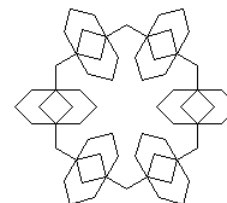
gyémánt 20



tömb1 10



tömb2 10



tömb3 10

A gyémánt egy hatszög, azonos hosszúságú oldalakkal. Sajnos nem szabályos hatszög, ezért a meg-rajzolása kicsit bonyolultabb.

```
eljárás gyémánt :méret
  ismétlés 2 [ismétlés 3 [előre :méret jobbra 45] jobbra 45]
vége
```

Az első két beágyazott változat szabályos hatszöget rajzol, amelynek minden oldalát három részre vágjuk, a középső harmad helyére illesztjük be megfelelő állásban a gyémántot.

```
eljárás tömb1 :méret
  ismétlés 6 [előre :méret balra 45 előre :méret jobbra 90
             gyémánt :méret
             előre :méret balra 45 előre :méret jobbra 60]
vége
```

```
eljárás tömb2 :méret
  ismétlés 6 [előre :méret balra 135 hátra :méret
             gyémánt :méret jobbra 90 előre :méret
             jobbra 45 előre :méret jobbra 60]
vége
```

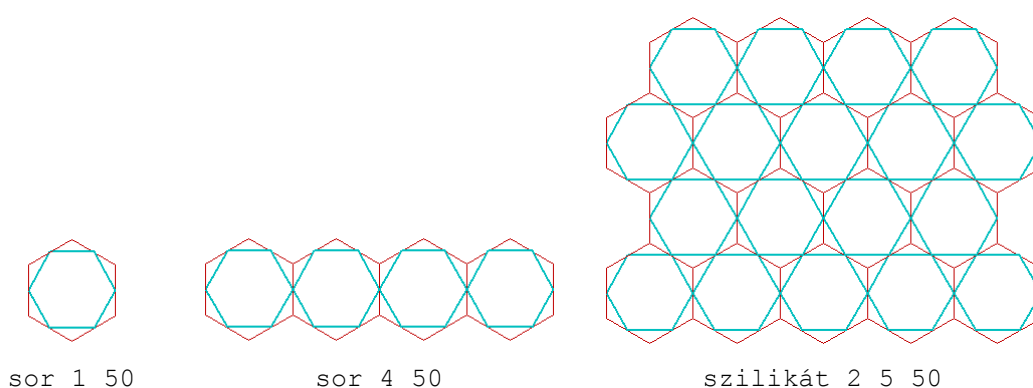
A harmadik beágyazott változat az első kettő kombinációja.

```
eljárás tömb3 :méret
  tömb1 :méret tömb2 :méret
vége
```

### Szilikát

Egyes ásványokban (pl. a szilikátok ilyenek) hatszög alapú rácsba rendeződnek el az atomok. Ha a térbeli rácsot síkban ábrázoljuk, akkor két egybefonódó hatszögrácsot látunk, ahol a belső hatszög oldalhossza a külső hatszög oldalhosszának gyök (2) / 3-szorosa.

Készíts Logo eljárást szilikát (szilikát :n :m :h) rajzolására, ahol a szilikátnak  $2 \cdot n$  sora és  $m$  oszlopa van, a nagyobb hatszög oldalhossza pedig  $h$ . Részfeladatként írd meg egy  $n$  :  $m$  :  $h$  eljárást, amely egysoros,  $m$  oszlopos szilikátot rajzol! A nagyobb hatszögek piros, a kisebbek pedig kétszeres vonalvastagságú, kék színűek legyenek!



Mivel a szilikátnak páros számú sora van és minden második sor eggyel kevesebb alapelemből áll, érdemes a mozaikot rajzoló eljárást dupla sorokra megírni. Itt célszerű (legalábbis a dupla eljárásnál) eltekinteni az állapotátlátszóságtól.

```
eljárás szilikát :n :m :h
  ismétlés :n [dupla :m :h] tollatfel hátra 3*:n tollatle
vége
```



```

eljárás dupla :m :h
  balra 60 előre :h jobbra 60 sor :m :h
  jobbra 60 előre :h balra 60 sor :m-1 :h
vége

```

A sormintában balról jobbra megrajzoljuk a piros, majd jobbról balra a kék hatszögeket.

```

eljárás sor :m :h
  tollvastagság! 1 tollszín! 4
  ismétlés :m [hatszög :h ismétlés 4 [előre :h jobbra 60]
    balra 240]
  balra 180 hátra :h/2 jobbra 30 tollvastagság! 2 tollszín! 3
  ismétlés :m [hatszög :h/2*gyök 3
    ismétlés 3 [előre :h/2*gyök 3 jobbra 60] balra 180]
  tollvastagság! 1 tollszín! 4 jobbra 150 előre :h/2
vége

```

```

eljárás hatszög :h
  ismétlés 6 [előre :h jobbra 60]
vége

```

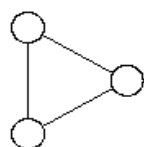
### Gráfrajzolásra épülő ásványok

A kristályok struktúráját sok esetben gráffal ábrázolják. A gráf csomópontokból és azokat valamilyen szabályszerűséggel összekötő élekből áll. A legtöbb esetben a gráf többféle csomópontot tartalmaz, azaz legalább kétféle alapelem rajzolására lesz szükségünk.

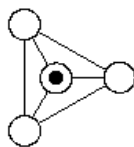
A legegyszerűbb esetekben a gráf egy sokszög, illetve elemek sokszögek csúcsaiba vagy oldalaira elhelyezve.

#### Szilikát – első változat

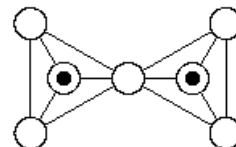
Egy szilikát ásvány ( $\text{Si}_2\text{O}_7$ ) háromszög alakban elhelyezkedő 3 oxigénatomból (10 sugarú kör), valamint térben egymás fölött elhelyezkedő egy szilícium (10 méretű fekete pötty) és egy oxigénatomból álló pár összekapcsolódásából (van egy közös oxigén) áll. Készíts Logo eljárásokat az alábbi ábrák megrajzolására, ahol a hosszabb vonal (kör középpontjától a másik kör középpontjáiig) :h hosszúságú, a rövidebb pedig ennek  $(\text{gyök } 3) / 3$ -szorosa!



három 100



alap 100



szilikát 100

Először rajzoljuk meg az oxigént (o) és a szilícium-oxigén párt (sio)!

```

eljárás o
  kör 10
vége

```

```

eljárás sio
  kör 10 tollvastagság! 10 pont tollvastagság! 1
vége

```

A kört a középpontjából kiindulva rajzoljuk:

```

eljárás kör :r
  tollatfel előre :r jobbra 90 tollatle
  ismétlés 360 [előre :r*3,14159/180 balra 1]
  tollatfel balra 90 hátra :r tollatle
vége

```

```

eljárás három :h
  ismétlés 3 [o él :h jobbra 120]
vége
    
```

Az élen a körökön belül felemelt tollal, a körök között leengedett tollal (rajzolva) megyünk végig. Az egyik eljárás az él végpontjában marad (él), a másik eljárás állapotátlátszó (élaa).

```

eljárás él :h
  tollatfel előre 10 tollatle előre :h-20 tollatfel előre 10 tollatle
vége
    
```

```

eljárás élaa :h
  él :h tollatfel hátra :h+20 tollatle
vége
    
```

Az alap a három eljárásra épül:

```

eljárás alap :h
  három :h jobbra 30 él :h/3*gyök 3 sio balra 60
  ismétlés 2 [élaa :h/3*gyök 3 jobbra 120]
  tollatfel előre :h/3*gyök 3 tollatle jobbra 150
vége
    
```

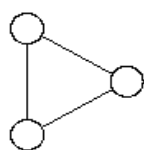
A szilikát két alap-ot tartalmaz:

```

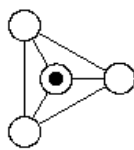
eljárásszilikát :h
  balra 120 alap :h jobbra 180 alap :h balra 60
vége
    
```

### Szilikát – második változat

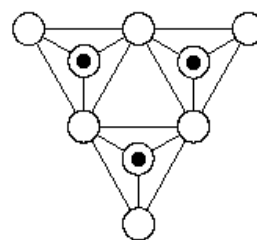
Egy szilikát ásvány ( $\text{Si}_3\text{O}_9$ ) háromszög alakban elhelyezkedő 3 oxigénatomból (10 sugarú kör), valamint térben egymás fölött elhelyezkedő egy szilícium (10 méretű fekete pötty) és egy oxigénatomból álló hármass összekapcsolódásából (vannak közös oxigének) áll. Készíts Logo eljárásokat az alábbi ábrák megrajzolására, ahol a hosszabb vonal (kör középpontjától a másik kör középpontjáiig) :h hosszúságú, a rövidebb pedig ennek  $(\text{gyök } 3) / 3$ -szoros!



három 100



alap 100



szilikát 100

A három és az alap eljárás megvalósítása azonos az előző feladattal. A szilikát pedig egy háromszög három csúcsába rajzolt alap-ból áll.

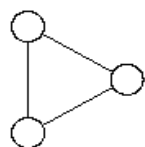
```

eljárás szilikát :h
  balra 150
  ismétlés 3 [alap :h él :h balra 120]
  jobbra 150
vége
    
```

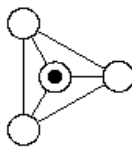
### Szilikát – harmadik változat

Egy szilikát ásvány ( $\text{Si}_6\text{O}_{18}$ ) háromszög alakban elhelyezkedő 3 oxigénatomból (10 sugarú kör), valamint térben egymás fölött elhelyezkedő egy szilícium (10 méretű fekete pötty) és egy oxigénatomból álló hatos összekapcsolódásából (vannak közös oxigének) áll. Készíts Logo eljárásokat az alábbi

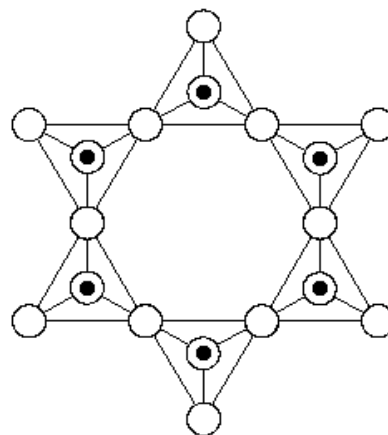
ábrák megrajzolására, ahol a hosszabb vonal (kör középpontjától a másik kör középpontjáig) :h hosszúságú, a rövidebb pedig ennek (gyök 3) /3-szorosa!



három 100



alap 100



szilikát 100

Itt is a szilikát eljárás különbözik az első szilikát rajzolásától. Most hatszög csúcaiba kell rajzolni az alap-ot.

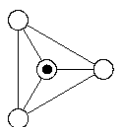
```

eljárás szilikát :h
  balra 150
  ismétlés 6 [alap :h él :h balra 60]
  jobbra 150
vége
    
```

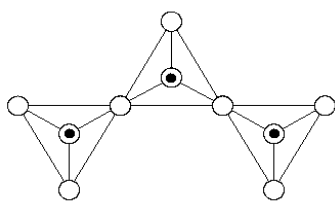
A következőkben az alapelemet nem egy sokszög csúcaira helyezzük, hanem sormintát építünk belőle.

### Piroxén

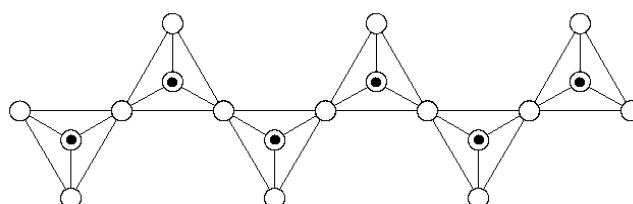
Polimer szilikátok szerkezetét mutatják az alábbi rajzok. Egy alapelem 4 molekulából áll az ábra szerinti elrendezésben. Az ábrán a háromszögek oldalhossza (kör középpontjától kör középpontjáig) :h, a belső vonalak hossza pedig :h/gyök 3.



alap 50



piroxén 3 50



piroxén 6 50

Készítsd el az alap :h és a piroxén :n :h eljárásokat, ahol :n az alapelemek száma a piroxénben!

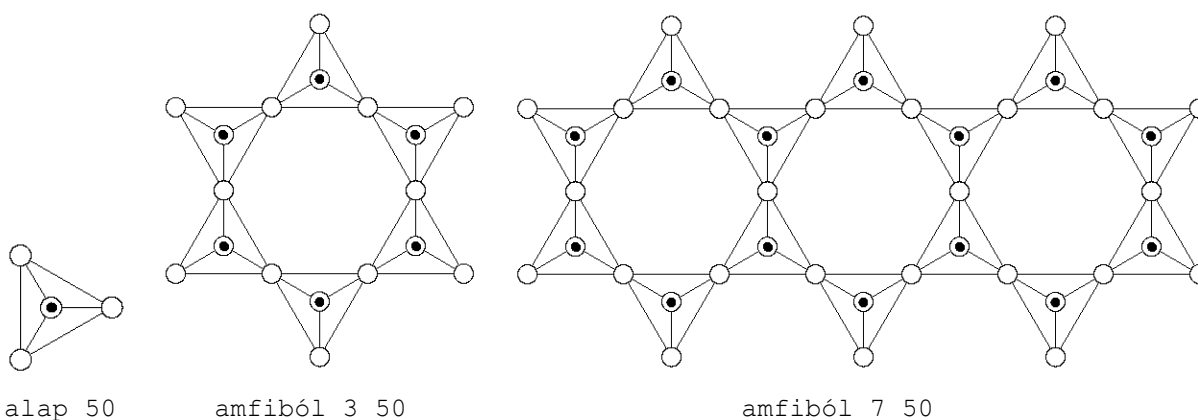
A sormintában ugyan egyforma elemek vannak, de felváltva más irányban kell őket rajzolnunk.

```

eljárás piroxén :n :h
  balra 30
  ismétlés :n/2 [alap :h jobbra 60 tollatfel előre 2*:h jobbra 120
                 tollatle alap :h tollatfel előre 2*:h jobbra 180
                 tollatle]
  hakülönben 1=maradék :n 2 [alap :h tollatfel balra 60 előre (:n-1)*:h]
                             [tollatfel balra 60 előre :n*:h]
  jobbra 60 tollatle
vége
    
```

### Amfiból

Polimer szilikátok szerkezetét mutatják az alábbi rajzok. Egy alapelem 4 molekulából áll (alap :h), az ábra szerinti elrendezésben. Az ábrán a háromszögek oldalhossza (kör középpontjától kör középpontjáig) :h, a belső vonalak hossza pedig :h/gyök 3.



Készítsd el az alap :h és az amfiból :n :h eljárásokat, ahol :n az alapelemek száma az amfiból mindkét sorában (:n biztosan páratlan)!

```

eljárás amfiból :n :h
  jobbra 30
  ismétlés :n/2 [tollatle alap :h jobbra 60 tollatfel előre :h
                 tollatle alap :h tollatfel előre :h balra 60]
  alap :h tollatfel előre 2*:h jobbra 180 tollatle
  ismétlés :n/2 [tollatle alap :h jobbra 60 tollatfel előre :h
                 tollatle alap :h tollatfel előre :h balra 60]
  alap :h jobbra 60 tollatle
vége
    
```

Megjegyzés: az amfiból két sor piroxénből áll, azaz ha a piroxén rajzolás nem lenne állapotátlátszó, akkor az első piroxén után 180 fokos elfordulással rajzolhatnánk a másodikat és az amfiból így elkészülne.

```

eljárás újamfiból :n :h
  balra 30
  újpiroxén :n :h jobbra 180 újpiroxén :n :h balra 150
vége

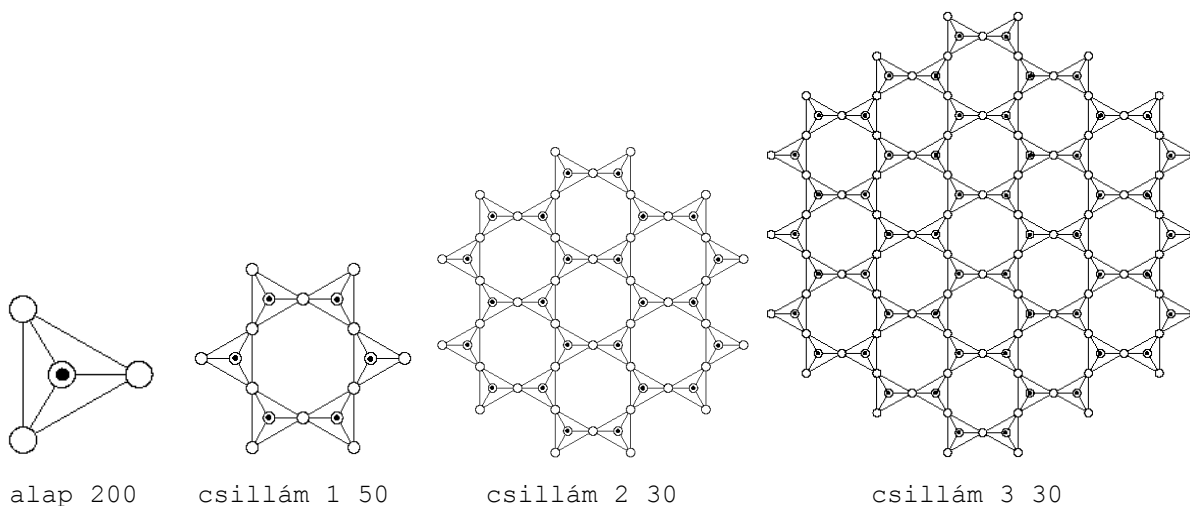
eljárás újpiroxén :n :h
  ismétlés :n/2 [alap :h jobbra 60 tollatfel előre 2*:h jobbra 120
                 tollatle alap :h tollatfel előre 2*:h jobbra 180
                 tollatle]
  alap :h jobbra 180
vége
    
```

### Csillám – első változat

Polimer szilikátok szerkezetét mutatják az alábbi rajzok. Egy alapelem 4 molekulából áll az ábra szerinti elrendezésben.

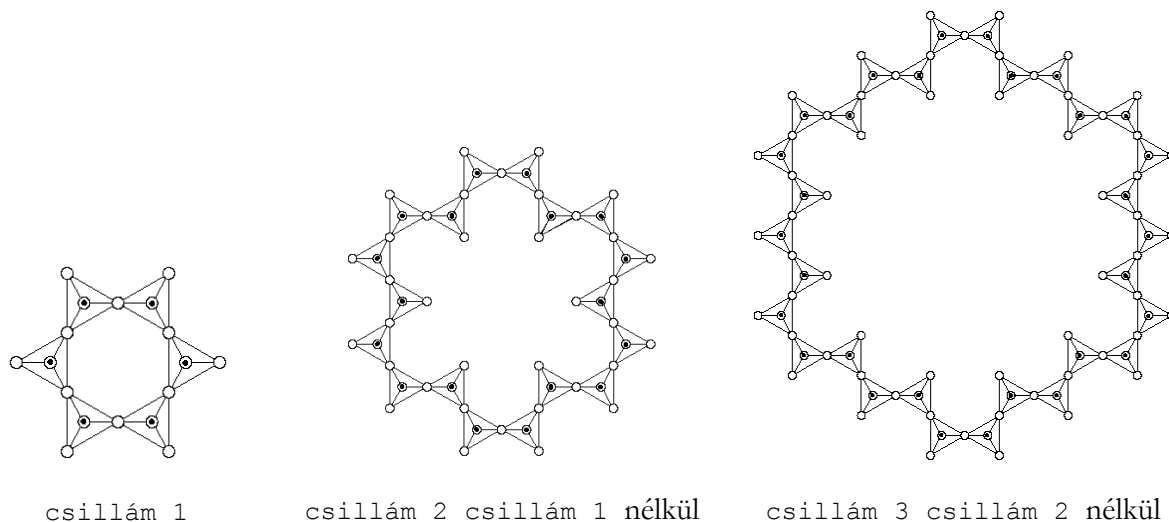
A csillám egyik változatában egyetlen hatszög alakú szerkezetből indulunk ki, a másik változatban pedig 3 hatszög alakúból.

Az ábrán a háromszögek oldalhossza (kör középpontjától kör középpontjáig) :h, a belső vonalak hossza pedig :h/gyök 3.



Készítsd el az alap :h és a csillám :n :h eljárásokat, ahol :n a csillám mérete, azaz a középső elemet vagy elemhármast tartalmazó körök száma (beleértve a kiinduló elemet is)!

Vizsgáljuk meg, miben különböznek az egymás utáni ábrák! Ebből látható, hogy az egyik megoldás egy rekurzív mozaik, ami egyre nagyobb hatszög alakban rajzolja egymás köré az alapelemeket.



eljárás csillám :n :h

ha :n>1 [ismétlés 6 [ismétlés :n-2 [darab 3 :h] darab 4 :h]]

hakülönben :n>2 [tollatfel hátra 2\* :h tollatle csillám :n-1 :h]

[tollatfel jobbra 60 hátra :h

ismétlés 6 [alap :h tollatfel jobbra 60 előre :h tollatle]]

vége

```

eljárás darab :db :h
ismétlés :db [alap :h tollatfel jobbra 60 előre :h tollatle]
jobbra 180
vége
    
```

Egy másik megoldás lehetne egy olyan mozaik rajzolása, amiben soronként más elemszámú csillám 1 van.

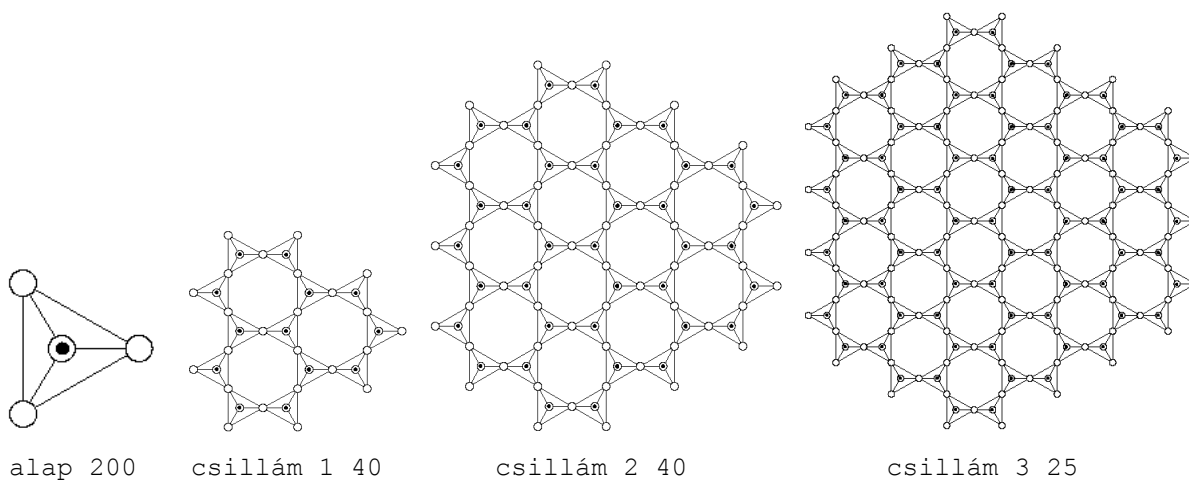
Ha az ábrákat 60 fokkal elforgatjuk, akkor látható, hogy csillám 2-ben 3 sor van, 2-3-2 elemmel, csillám 3-ban pedig 5 sor van, 3-4-5-4-3 elemmel. Azaz általánosan csillám :n-ben :n\*2-1 sor van, :n-:n+1-...-:n+1-:n elemmel.

### Csillám – második változat

Polimer szilikátok szerkezetét mutatják az alábbi rajzok. Egy alapelem 4 molekulából áll az ábra szerinti elrendezésben.

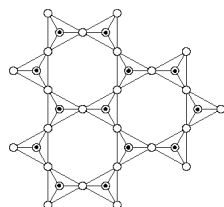
A csillám egyik változatában egyetlen hatszög alakú szerkezetből indulunk ki, a másik változatban pedig 3 hatszög alakúból.

Az ábrán a háromszögek oldalhossza (kör középpontjától kör középpontjáig) :h, a belső vonalak hossza pedig :h/gyök 3.

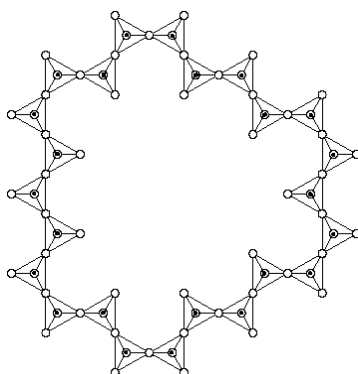


Készítsd el az alap :h és a csillám :n :h eljárásokat, ahol :n a csillám mérete, azaz a középső elemet vagy elemhármast tartalmazó körök száma (beleértve a kiinduló elemet is)!

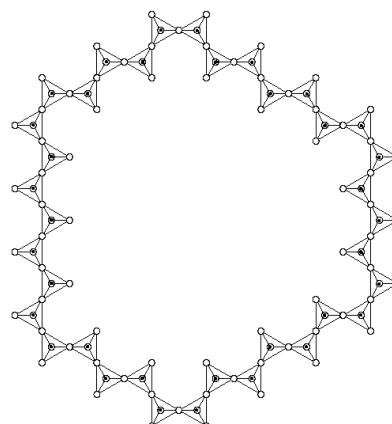
Itt is megnézhetjük, miben különböznek az egymás utáni ábrák! Ebből látható, hogy az egyik megoldás egy rekurzív mozaik, ami egyre nagyobb, de nem szabályos hatszög alakban rajzolja egymás köré az alapelemeket.



csillám 1



csillám 2 csillám 1 nélkül



csillám 3 csillám 2 nélkül

```

eljárás csillám :n :h
  ha :n>1 [ismétlés 3 [ismétlés :n-1 [darab 3 :h] darab 4 :h
            ismétlés :n-2 [darab 3 :h] darab 4 :h]]
  hakülönben :n>2 [tollatfel hátra 2*:h tollatle csillám :n-1 :h]
                  [tollatfel hátra 2*:h tollatle darab 5 :h darab 5 :h
                    darab 5 :h jobbra 180 alap :h]

```

vége

```

eljárás darab :db :h
  ismétlés :db [alap :h tollatfel jobbra 60 előre :h tollatle] jobbra 180
vége

```

Egy másik megoldás lehetne egy olyan mozaik rajzolása, amiben soronként más elemszámú csillám 1 van.

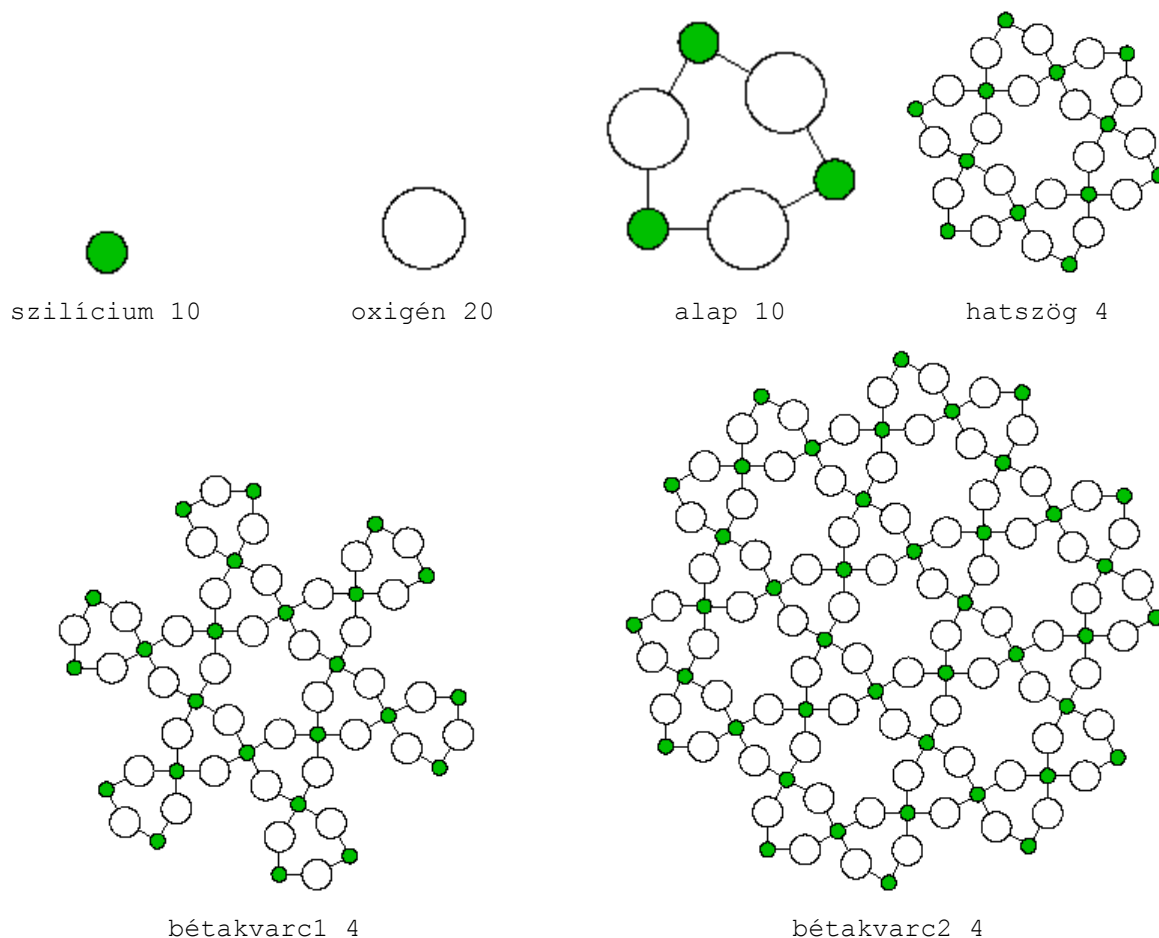
Ha az ábrákat 60 fokkal elforgatjuk, akkor látható, hogy csillám 2-ben 4 sor van, 2-3-4-3 elemmel, csillám 3-ban pedig 6 sor van, 3-4-5-6-5-4 elemmel. Azaz általánosan csillám :n-ben :n\*2 sor van, :n-:n+1-...-:n+1 elemmel.

### Béta-kvarc

A Béta-kvarc rácstruktúrája síkra vetíthető. Ebben az ásványban egy szilíciumatom négy oxigénatomhoz, illetve minden oxigénatom két szilíciumatomhoz kapcsolódik.

Készíts Logo eljárásokat (bétakvarc1 :r, bétakvarc2 :r) a Béta-kvarc megrajzolására! Ehhez a következő eljárásokat használd:

- szilícium :r :r sugarú, zöld színű kör a szilíciumatom képének;
- oxigén :r :r sugarú, üres kör az oxigénatom képének;
- alap :r 3 oxigén- és 3 szilíciumatomból álló struktúra, a szilíciumatom :r, az oxigénatom pedig 2\* :r sugarú, a köztük levő kötéseket jelző szakaszok pedig 2\* :r hosszúak;
- hatszög :r 6 alapelemből felépülő struktúra, hatszög alakban;
- bétakvarc1 :r a hatszög szélein levő szilíciumatomokból egy-egy újabb alapelem nő ki;
- bétakvarc2 :r a hatszög szélein levő szilíciumatomokból egy-egy újabb hatszög nő ki;



```

eljárás szilícium :r
  üreskör :r töltőszín! 2 tollatfel balra 90 előre :r tölt
  hátra :r jobbra 90 tollatle
vége

```

```

eljárás üreskör :r
  ismétlés 360 [előre :r*3,14159/180 balra 1]
vége

```

```

eljárás oxigén :r
  üreskör :r
vége

```

Az alap a gráf alapeleme, egy egyszerű háromszögre épített gráf.

```

eljárás alap :r
  ismétlés 3 [szilícium :r körív :r 90 jobbra 90 előre 2*:r jobbra 90
              oxigén 2*:r körív 2*:r 150 jobbra 90 előre 2*:r jobbra 90]
vége

```

```

eljárás körív :r :fok
  ismétlés :fok [előre :r*3,14159/180 balra 1]
vége

```

Az hatszög egy hatszögre épített alap-okból álló gráf.

```

eljárás hatszög :r
  ismétlés 6 [alap :r jobbra 90 előre 2*:r jobbra 90 körív 2*:r 210
              jobbra 90 előre 2*:r jobbra 90 körív :r 90]
vége

```



A bétakvarc1 a hatszög csúcsaira egy-egy alap-ot tesz.

```

eljárás bétakvarc1 :r
hatszög :r
ismétlés 6 [körív :r 90 jobbra 90 előre 2*:r jobbra 90
körív 2*:r 150 jobbra 90 előre 2*:r jobbra 90
körív :r 180 alap :r körív :r 270 jobbra 90
előre 2*:r jobbra 90 körív 2*:r 150 jobbra 90
előre 2*:r jobbra 90 körív :r 180]

```

vége

A bétakvarc2 a hatszög csúcsaira egy-egy hatszög-et tesz.

```

eljárás bétakvarc2 :r
hatszög :r
ismétlés 6 [körív :r 90 jobbra 90 előre 2*:r jobbra 90
körív 2*:r 150 jobbra 90 előre 2*:r jobbra 90
körív :r 180 hatszög :r körív :r 270 jobbra 90
előre 2*:r jobbra 90 körív 2*:r 150 jobbra 90
előre 2*:r jobbra 90 körív :r 180]

```

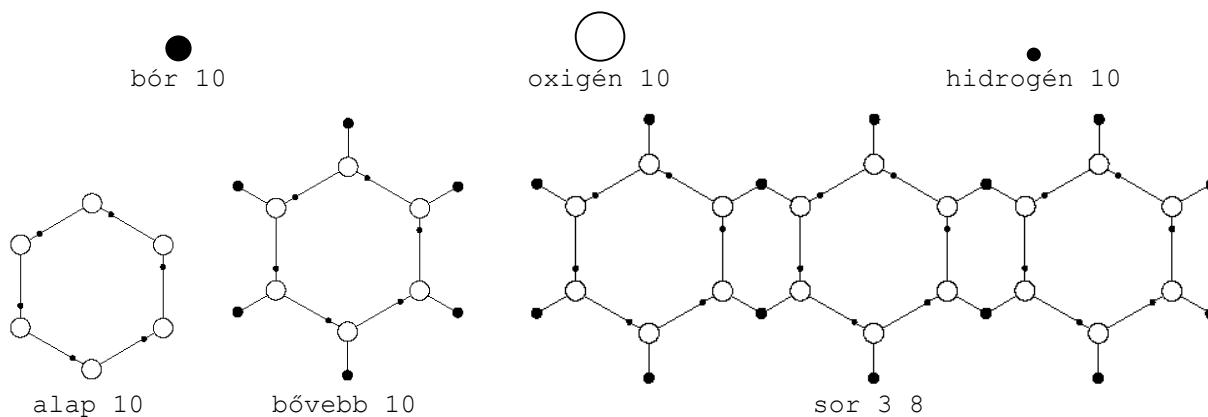
vége

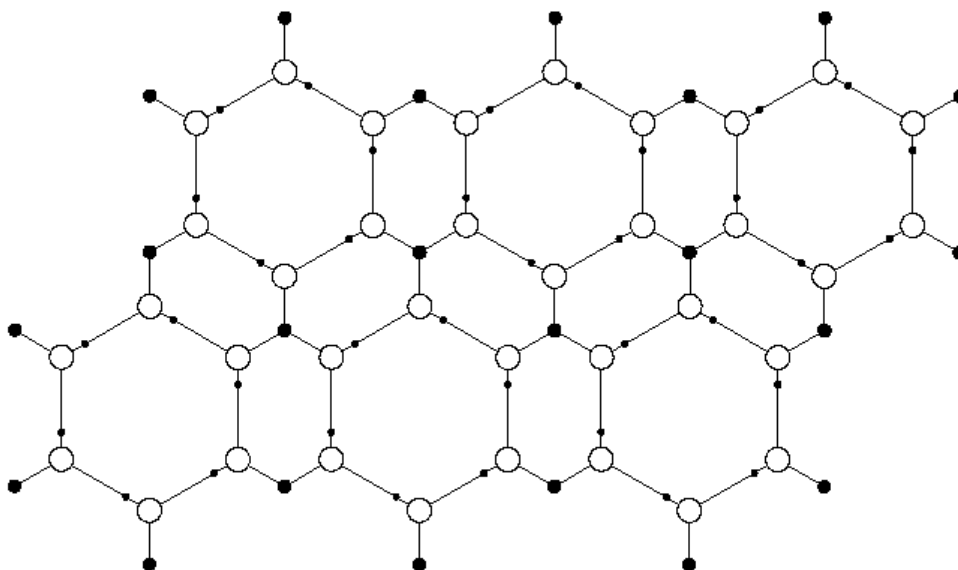
### Sassolin

A sassolin (B(OH)<sub>3</sub>) rácstruktúrája síkra vetíthető. Ebben az ásványban minden bóratomot (:r/2 sugarú festett kör) három oxigénatom (:r sugarú üres kör) vesz körül. Az oxigénatomokhoz két hidrogén- (:r/4 sugarú festett kör) és egy bóratom tartozik. A bór és az oxigén távolsága legyen 3\*:r, a hidrogén az egyik oxigén szomszédjától :r, a másiktól pedig 5\*:r távolságra van.

Készíts Logo eljárást (sassolin :m :n :r) a sassolin megrajzolására! Ehhez a következő eljárásokat használd:

bór :r	bóratom, :r/2 sugarú festett kör
oxigén :r	oxigénatom, :r sugarú üres kör
hidrogén :r	hidrogénatom, :r/4 sugarú festett kör
alap :r	6 oxigén- és hidrogénatomból álló struktúra, hatszög alakban;
bővebb :r	6 oxigén-, bór- és hidrogénatomból álló struktúra, hatszög alakban;
sor :n :r	:n darab bővebb :r struktúra egymás mellé helyezve;
sassolin :m :n :r	:m darab sor :n :r struktúra egymás fölé helyezve;





sassolin 2 3 8

A sassolin egy mozaik, ami egyszerű sorokból áll.

```

eljárás sassolin :m :n :r
  ismétlés :m [sor :n :r tollatfel előre :r jobbra 60 előre 4,5*:r
    balra 60 előre 4,5*:r jobbra 60 előre 8,5*:r
    balra 60 előre 7,5*:r tollatle]
vége
  
```

A sorok bonyolultságát az állapotátlátszóság miatti visszalépés okozza.

```

eljárás sor :n :r
  ismétlés :n-1 [bővebb :r tollatfel előre :r jobbra 60
    előre 4,5*:r jobbra 60 előre 4,5*:r balra 60
    előre 8,5*:r jobbra 60 előre 8,5*:r balra 120
    hátra :r tollatle]
  bővebb :r tollatfel
  ismétlés :n-1 [előre :r balra 60 előre 8,5*:r balra 60
    előre 8,5*:r jobbra 60 előre 4,5*:r balra 60
    előre 4,5*:r jobbra 120 hátra :r]
  tollatle
vége
  
```

A bővebb kirajzol egy hatos szimmetriájú részt.

```

eljárás bővebb :r
  ismétlés 6 [jobbra 90 körív :r 120 jobbra 90 előre 3*:r
    jobbra 90 telikör :r/2 balra 90 hátra 3*:r balra 90
    körív :r 480 jobbra 90 előre 5*:r jobbra 90 telikör :r/4
    körív :r/4 180 jobbra 90 előre :r]
vége
  
```

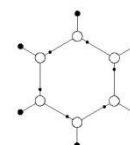
vége

```

eljárás körív :r :f
  ismétlés :f [előre :r*3,14159/180 balra 1]
vége
  
```

```

eljárás telikör :r
  ismétlés 360 [előre :r*3,14159/180 balra 1]
  tollatfel balra 90 előre :r tölt hátra :r jobbra 90 tollatle
vége
  
```



## Rekurzió

Rekurzió alkalmazására sokszor szükség lehet, hagyományosnak tűnő ábrák rajzolásánál is. A Logo nyelv ciklusa ugyanis arra alkalmas elsősorban, hogy azonos tevékenységeket végezzen sokszor.

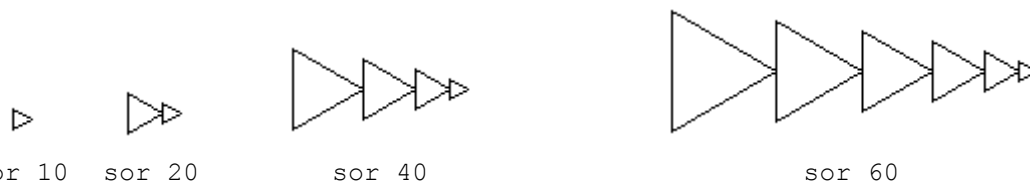
### Sorminta változó méretű elemekkel

Az egyik legegyszerűbb eset, amikor sormintákat rajzolunk, de rajzolás közben változik a sorminta elemeinek mérete.

#### Csökkenő háromszögek

Egy sorminta egyforma elemekből (háromszögekből) épül fel. A sorminta utolsó eleme 10 egység oldalhosszúságú háromszög, ami elemenként 10 egységgel növekszik.

Készíts Logo eljárást (`sor :h`), amely az alábbi sormintát rajzolja!



```
eljárás sor :h
  hátra :h/2 háromszög :h
  ha :h>0 [jobbra 60 előre :h balra 60 sor :h-10]
vége

eljárás háromszög :h
  ismétlés 3 [előre :h jobbra 120]
vége
```

#### Távolodó madarak

Rajzold le az égen egymás után szálló madarakat (`madarak :db :méret :arány`). Az első a legnagyobb, majd a következő arányosan mindig kisebb. A `:db` paraméter a madarak számát, a `:méret` az első madár nagyságát, az `:arány` pedig a kicsinyítés mértékét adja meg.



```
eljárás madár :méret
  jobbra 30 ívj :méret balra 60 ívb :méret jobbra 30
vége
```

A szárnyak 60 fokos körívekből állnak.

```

eljárás ívj :r
  ismétlés 60 [előre :r*2*3,14/360 jobbra 1]
  tollatfel ismétlés 60 [balra 1 hátra :r*2*3,14159/360] tollatle
vége

eljárás ívb :r
  ismétlés 60 [előre :r*2*3,14/360 balra 1]
  tollatfel ismétlés 60 [jobbra 1 hátra :r*2*3,14159/360] tollatle
vége

```

Ne használjunk ciklust, hiszen a madarak mérete lépésről-lépésre csökken.

```

eljárás madarak :db :méret :arány
  ha :db>0 [madár :méret tollatfel jobbra 30
            előre :méret*1,5 balra 30 tollatle
            madarak :db-1 :méret*:arány :arány]
vége

```

## Jégcsap

Egy jégcsap a középső részére mindig szimmetrikusan nő (bal oldali ábra). A jégcsap egyik felét úgy kell megrajzolni, hogy a jégcsapban levő jég mennyiségének negyed része van a leghosszabb oszlopban, a háromnegyed része a többiben. A további részre ugyanez teljesül, azaz annak is negyed része van a leghosszabb oszlopban, a többi pedig a maradékban, és így tovább, amíg 1 egységnyit nem kapunk.

Készíts Logo eljárást (jégcsap :x), amely :x mennyiségű jégből álló jégcsapot rajzol a képernyőre!



jégcsap 1000

baloldali fél jégcsap

jobboldali fél jégcsap

A jégcsap két fél jégcsapból áll, amelyek egymás tükörképei. Így használható rájuk a tükrözési szabály.

```

eljárás jégcsap :x
  jobbra 180 féljég :x/2 1
  jobbra 90 hátra 1 balra 90 féljég :x/2 (-1)
vége

```

A fél jégcsap egy sorminta, ami egyre kisebb elemekből áll. A negyed rész a függőleges vonal hossza, a háromnegyed pedig a további – rekurzív – részé.

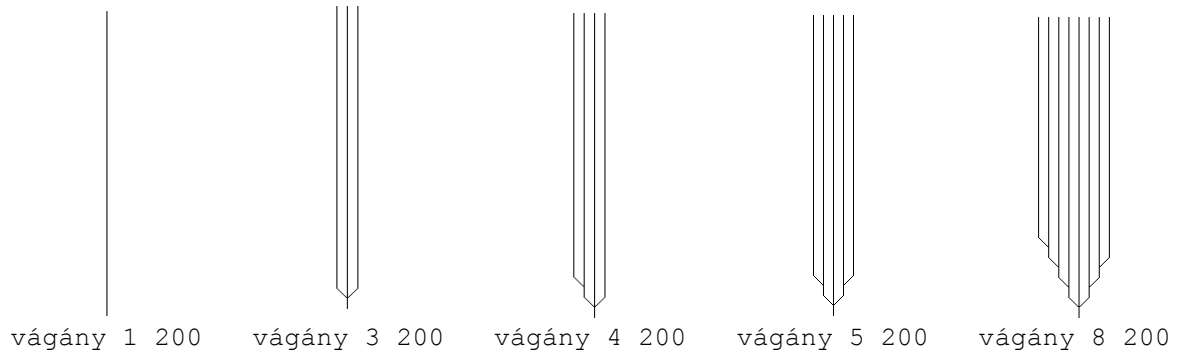
```

eljárás féljég :x :i
  előre :x/4 hátra :x/4
  ha :x>1 [jobbra 90 előre :i balra 90 féljég 3*:x/4 :i
          jobbra 90 hátra :i balra 90]
vége

```

## Vágányok

Készíts Logo eljárást (vágány :n :h), amely egy :n vágányból álló vasútállomás szerkezetét rajzolja ki az alábbi formában (:h a leghosszabb vágány hossza, jobbra, illetve balra haladva a vágányok 10-10 egységgel lesznek rövidebbek):



Két rekurzív vágányrajzoló eljárást írunk, az egyik az egyenestől balra, a másik pedig jobbra rajzolja a vágányokat.

```

eljárás vágány :n :h
  előre :h hátra :h-10
  ha :n>1 [vágány1 :h-10 egészhányados :n 2]
  ha :n>2 [vágány2 :h-10 egészhányados :n-1 2]
  hátra 10
vége
  
```

```

eljárás vágány1 :h :n
  balra 45 előre 10*gyök 2 jobbra 45 előre :h-10 hátra :h-20
  ha :n>1 [vágány1 :h-20 :n-1]
  hátra 10 balra 45 hátra 10*gyök 2 jobbra 45
vége
  
```

```

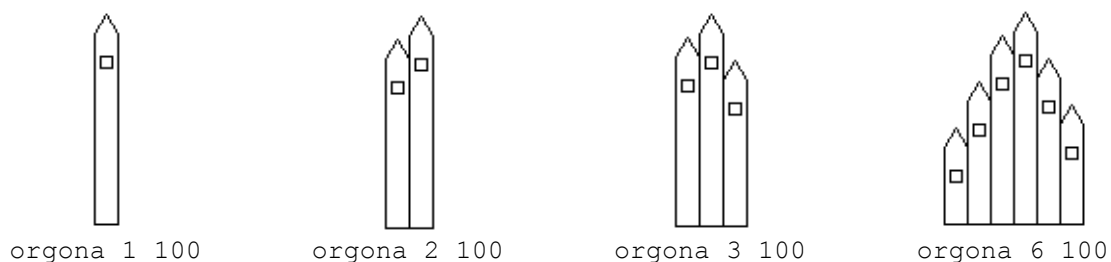
eljárás vágány2 :h :n
  jobbra 45 előre 10*gyök 2 balra 45 előre :h-10 hátra :h-20
  ha :n>1 [vágány2 :h-20 :n-1]
  hátra 10 jobbra 45 hátra 10*gyök 2 balra 45
vége
  
```

Megjegyzés: Itt is elég lenne egyetlen rekurzív eljárás, mert a bal-, illetve a jobboldali vágányok egymás tükörképei, azaz használhatnánk a tükrözési szabályt.

## Orgonasíp

Az orgonasípot speciális módon szokták elrendezni. Középen van mindig a leghosszabb síp. Tőle balra található a második, jobbra pedig a harmadik leghosszabb. A következőt megint a bal szélre rakják, az azt követőt pedig a jobb szélre, ... és így tovább.

Készíts Logo eljárást (orgona :db :h), amely a mintának megfelelő orgonát rajzolja, :db sípból áll, s a leghosszabb hossza :h!



Az orgonasípok csak abban különböznek a vágányoktól, hogy balra és jobbra más méretű orgonasípkat kell rajzolni.

```
eljárás orgona :n :h
  síp :h 10
  ha :n>1 [balorgona (:n-1)/2 :h-10]
  ha :n>2 [jobborgona (:n-2)/2 :h-2*10]
vége
```

Az orgona bal oldali részét rajzolja meg.

```
eljárás balorgona :n :h
  tollatfel balra 90 előre 10 jobbra 90 tollatle síp :h 10
  ha :n>1 [balorgona :n-1 :h-2*10]
  tollatfel balra 90 hátra 10 jobbra 90 tollatle
vége
```

Az orgona jobb oldali részét rajzolja meg.

```
eljárás jobborgona :n :h
  tollatfel jobbra 90 előre 10 balra 90 tollatle síp :h 10
  ha :n>1 [jobborgona :n-1 :h-2*10]
  tollatfel jobbra 90 hátra 10 balra 90 tollatle
vége
```

```
eljárás síp :h :k
  előre :h jobbra 30 előre :k jobbra 120 előre :k jobbra 30
  előre :h jobbra 90 előre :k jobbra 90
  tollatfel előre :h-:k jobbra 90 előre :k/4 tollatle
  ismétlés 4 [előre :k/2 jobbra 90]
  tollatfel hátra :k/4 balra 90 hátra :h-:k tollatle
vége
```

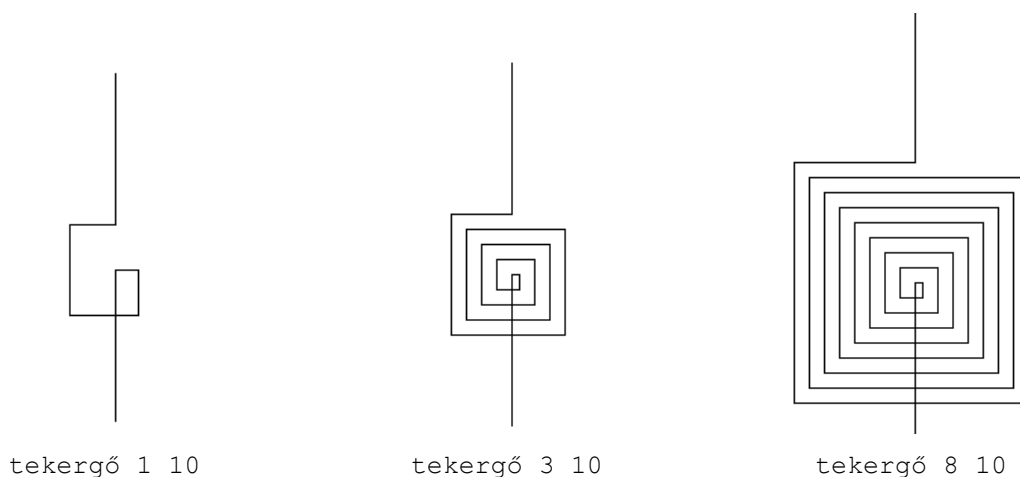
## Spirálok

A spirálok készítése a sokszögek rajzolásából indul ki – a spirál egy olyan sokszög, amelynek az oldalhossza oldalanként szabályosan változik (a programozó elképzelésétől függően nő vagy csökken).

## Tekergő

Egy drótszálat spirál alakban tekertek fel az ábrának megfelelő módon.

Készíts Logo eljárást (tekergő :db :h), amely egy :db-szer feltekert drótot rajzol, amelynél a legrövidebb egyenes drótdarab hossza :h, a drót két szélső darabja hossza pedig  $10 * :h$ !



A drót egyszeri körbetekerése 4–szer hívja meg a spirált, tehát minden teljes fordulat 4-szeres hívást jelent. A plusz 1 hívás azért kell, mert nem teljes fordulatot teszünk meg, hanem többet.

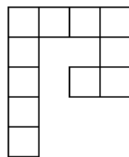
```
eljárás tekergő :db :h
  előre 10* :h spirál 1+4* :db :h :h előre 10* :h
vége
```

```
eljárás spirál :db :h :növény
  ha :db > 0 [előre :h jobbra 90 spirál :db-1 :h+ :növény :növény]
  [előre :h/2 balra 90]
vége
```

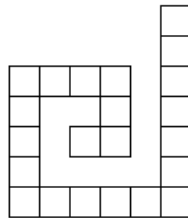
### Spirál négyzetekből

Spirált nemcsak vonalakkból, hanem más alakzatokból, például négyzetekből is készíthetünk. A spirál rajzolás szabálya ekkor az, hogy a spirál ágai mindig az előzőnél eggyel több négyzetből állnak, s az ágak végén 90 fokot kell fordulni.

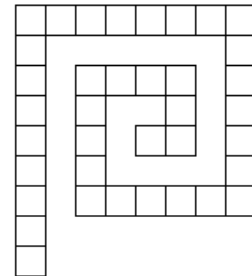
Készíts Logo eljárást (`spirál :db :h`), amely `:h` oldalhosszú négyzetekből `:db` ágú spirált rajzol!



spirál 4 10



spirál 6 10



spirál 8 10

Mivel a spirál egyes ágaira rajzolható négyzetek száma a spirál eredeti ágai számától függ, ezért érdemes egy belső eljárást írni, aminek még egy paraméter adunk – az adott ágra rajzolható négyzetek számát.

```
eljárás spirál :db :h
  spi :h :db 2
vége
```

Az egyes spirálágaknál a legegyszerűbb megvalósítási mód, ha a forduláskor levő négyzetet mindkét ágra megrajzoljuk. Most célszerű a sorok közös részét – pl. az első sor 2, a második 3 négyzetből áll – kétszer megrajzolni, egyszerűbb lesz így a rekurzív eljárás.

```
eljárás spi :h :db :n
  ismétlés :n [négyzet :h jobbra 90 előre :h balra 90] balra 90
  ha :db > 1 [spi :h :db-1 :n+1]
vége
```

```
eljárás négyzet :h
  ismétlés 4 [előre :h jobbra 90]
vége
```

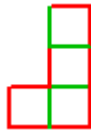
### Színes spirál

Zöld négyzetekből és piros vonalakkal spirált építhetünk. A spirál egyes szakaszain egyre több zöld négyzetet kell rajzolni úgy, hogy a spirál vonala piros legyen!

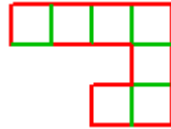
Készíts Logo eljárást (`spirál :n :h`) a spirál megrajzolására!



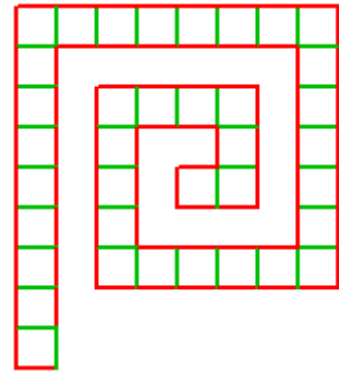
spirál 1 10



spirál 2 10



spirál 3 10



spirál 8 10

Ez a feladat tulajdonképpen az előző feladatbeli spirál vonalainak megfelelő kiszínezése. Itt a közös elemeket, azaz a sarkokat, csak egyszer szabad megrajzolni, a színezésük különben rossz lenne.

```
eljárás spirál :n :h
  tollszín! 12 ismétlés 2 [előre :h jobbra 90]
  tollszín! 2 előre :h jobbra 90 tollszín! 12 előre :h hátra :h
  jobbra 90 spi :n :h 1
vége
```

Az előkészítés után jöhet a rekurzív spirál rajzolás.

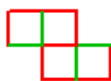
```
eljárás spi :n :h :i
  sor :i :h tollszín! 2
  ha :i<:n [spi :n :h :i+1]
vége
```

```
eljárás sor :n :h
  ismétlés :n-1 [ismétlés 2 [tollszín! 2 előre :h tollszín! 12
    jobbra 90 előre :h jobbra 90]
    tollszín! 12 jobbra 90 előre :h balra 90]
  tollszín! 2 ismétlés 2 [előre :h jobbra 90]
  tollszín! 12 ismétlés 2 [előre :h jobbra 90]
  tollszín! 2 ismétlés 2 [előre :h jobbra 90] jobbra 90
vége
```

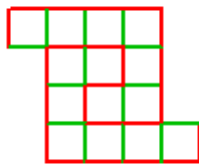
### Kettős spirál

Zöld négyzetekből és piros vonalokból kettős (vagy más néven dupla) spirált építhetünk. A spirál egyes szakaszain egyre több zöld négyzetet kell rajzolni úgy, hogy a spirál vonala piros legyen!

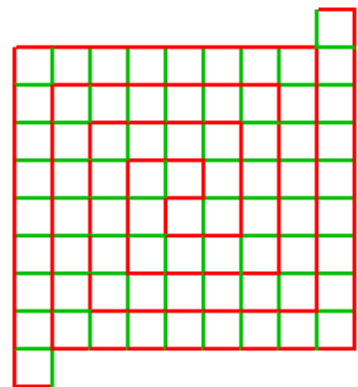
Készíts Logo eljárást (duplaspirál :n :r) a kettős spirál megrajzolására!



duplaspirál 1 10



duplaspirál 3 10



duplaspirál 8 10



A duplaspirálban elvileg párhuzamosan kétfelé indulunk a rajzolással, de ezt természetesen egymás után is megtehetjük. Ehhez persze biztosítani kell, hogy az első spirál megrajzolása után pontosan a kezdőállapotba álljunk vissza!

```
eljárás duplaspirál :n :h
  dupla :n :h xhely yhely irány
vége
```

Kettős spirált rajzol ki.

```
eljárás dupla :n :h :x :y :i
  tollvastagság! 2 tollszín! 12 ismétlés 2 [előre :h jobbra 90]
  tollszín! 2 előre :h jobbra 90
  tollszín! 12 előre :h hátra :h jobbra 90 spirál :n :h 1
  tollatfel xyhely! :x :y irány! :i előre :h jobbra 90 előre :h
  balra 90 előre :h balra 180 tollatle
  tollszín! 12 ismétlés 2 [előre :h jobbra 90]
  tollszín! 2 előre :h jobbra 90
  tollszín! 12 előre :h hátra :h jobbra 90
  spirál :n :h 1 tollvastagság! 1
  tollatfel xyhely! :x :y irány! :i tollatle
vége
```

Spirál rajzolása.

```
eljárás spirál :n :h :i
  sor :i :h
  ha :i<:n [spirál :n :h :i+1]
vége
```

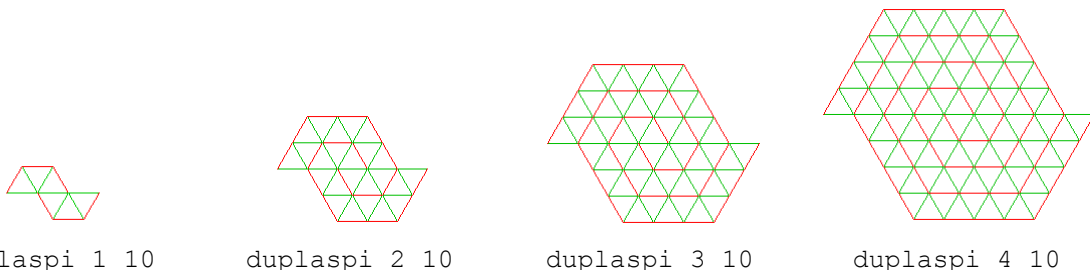
Egy sor négyzetet rajzol ki.

```
eljárás sor :n :h
  ismétlés :n-1 [ismétlés 2 [tollszín! 2 előre :h tollszín! 12
    jobbra 90 előre :h jobbra 90]
    jobbra 90 előre :h balra 90]
  tollszín! 2 ismétlés 2 [előre :h jobbra 90]
  tollszín! 12 ismétlés 2 [előre :h jobbra 90]
  tollszín! 2 ismétlés 2 [előre :h jobbra 90] jobbra 90
vége
```

### Kettős spirál háromszögekből

Zöld egyenlő oldalú háromszögekből és piros vonalokból kettős spirált építhetünk. A spirál egyes szakaszain egyre több, piros alapú zöld háromszöget kell rajzolni.

Készíts Logo eljárást (duplaspi :n :r) a kettős spirál megrajolására!



A kettős spirál két ága abban különbözik egymástól, hogy kicsit más helyen kezdődnek és ellentétes irányban indulnak. Mivel ugyanazt az eljárást kell végrehajtanunk, legyen a megoldás valóban a párhuzamos rajzolás, azaz alkalmazzunk két teknőcöt!

```

eljárás duplaspi :n :h
  újteknőc "teki (lista xhely yhely irány)
  figyelj [teki] tollatfel balra 90 előre :h balra 90 tollatle
  figyelj [0 teki] balra 30 spirál :n :h 1
vége

```

```

eljárás spirál :n :h :i
  ismétlés 3 [sor :i :h balra 60]
  ha :i<:n [spirál :n :h :i+1]
vége

```

```

eljárás sor :n :h
  ismétlés :n [tollszín! 12 előre :h tollszín! 2
               ismétlés 2 [balra 120 előre :h]
               tollszín! 12 balra 120 előre :h]
vége

```

### Spirál betűkből

Spirált nemcsak vonalakkból, hanem más alakzatokból, például betűkből is készíthetünk. A spirál rajzolás szabálya ekkor az, hogy a spirál ágai mindig az előzőnél eggyel több betűből állnak, s az ágak végén 90 fokot kell fordulni.

Készíts Logo eljárást (`spirál :db :h :szó`), amely a `:szó` betűiből (`:h` méretű betűkből) `:db` ágú spirált rajzol!

<pre> T C Ő N E   K K  T E N Ő </pre>	<pre>           N           K T C Ő N E E   K T K  T E C N      Ő Ő C T E K N </pre>	<pre> L O G O L O G O           L         G O L O       O   G G     L  L O O     O   L   G O L O G O </pre>
---------------------------------------	--	---

spirál 4 15 "TEKNŐC

spirál 6 15 "TEKNŐC

spirál 7 15 "LOGO

A `spirál` eljárás az eredeti paraméterezettséggel készült el. Feladata a `spi` meghívása, amely két segédparamétert tartalmaz.

```

eljárás spirál :db :h :szó
  tollatfel spi :h :db :szó 1 0 tollatle
vége

```

A `spi` eljárás kiírja egy sorba a betűket, majd fordul és folytatja tovább. Legyen `:n` az adott sorba írandó betűk száma, `:sor` pedig az összes eddig kiírt betű száma!

```

eljárás spi :h :db :szó :n :sor
  karakterek :n :sor :h :szó balra 90
  hakülönben :db>1 [spi :h :db-1 :szó :n+1 :sor+:n]
                  [karakter :sor+:n :szó]
vége

```

A karakterek karakterenként írja ki a betűsorozatokat.

```

eljárás karakterek :n :sor :h :szó
  karakter :sor :szó jobbra 90 előre :h balra 90
  ha :n>1 [karakterek :n-1 :sor+1 :h :szó]
vége

```

```

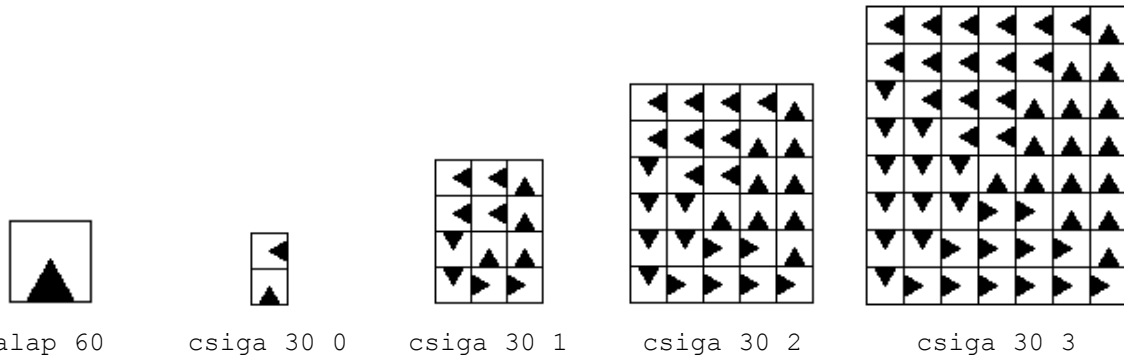
eljárás karakter :sor :szó
  betűzd elem 1+maradék :sor elemszám :szó :szó
vége

```

## Csigavonal

Egy csigavonalat az alábbi alap :h ábrából rajzolunk.

Készítsd el az alap :h és a csiga :h :db eljárásokat, ahol :db jelentése: a csigavonal az első két lépés után (a 30 0 paraméterű ábra) :db darab teljes kört tesz meg, a bal felső sarokban végződve!



eljárás alap :h

```
ismétlés 4 [előre :h jobbra 90]
jobbra 90 előre :h/5 balra 60
ismétlés 2 [előre 3*:h/5 jobbra 120]
hátra :h/5 előre :h/2 tollatfel jobbra 90 előre :h/5 tollatle
tölt tollatfel hátra :h/5 balra 90 előre :h/2 tollatle jobbra 90
vége
```

eljárás csiga :h :n

```
alap :h előre :h balra 90 hátra :h alap :h előre :h
ha :n>0 [oldalak :h 2*:n-1 1]
vége
```

Vegyük észre, hogy a csigavonal rajzolása közben, minden második fordulat után eggyel nő a rajzolható alapelemek száma!

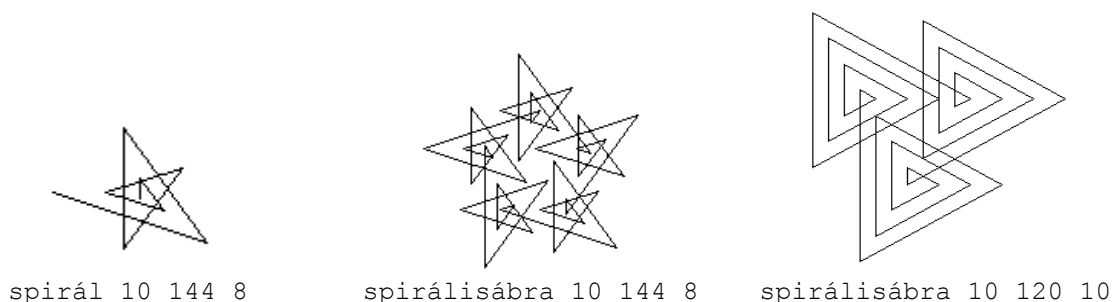
eljárás oldalak :h :n :db

```
alap :h balra 90 ismétlés :db [alap :h előre :h]
alap :h balra 90 ismétlés :db [alap :h előre :h]
ha :n>0 [oldalak :h :n-1 :db+1]
vége
```

## Spirál-sokszög

Úgynevezett spiráloldalú sokszöget úgy rajzolhatunk, hogy a sokszög rajolásában szereplő elmozdulást és elfordulást egyetlen spirálrajzoló eljárás hívásával helyettesítjük.

Készíts Logo eljárást (spirálisábra :hossz :szög :db), amely spirálisoldalú sokszöget rajzol! Egy spirált a spirál :hossz :szög :db eljárás rajzoljon! A spirál legrövidebb oldala és annak növekménye :hossz, elfordulás-szöge :szög, oldalai száma pedig :db legyen!



```

eljárás spirál :h :f :db
  spirális :h :f :db 1
vége

eljárás spirális :h :f :max :db
  előre :h*:db jobbra :f
  ha :db<:max [spirális :h :f :max :db+1]
vége

```

Annyiszor kell ismételn a spirális ábrát, amíg vissza nem térünk a kiinduló állapotba, azaz a fordulatok összege 360 fok többszöröse nem lesz.

```

eljárás spirálisábra :h :f :max
  ismétlés keres maradék :max*:f 360 1 [spirál :h :f :max]
vége

eljárás keres :f :n
  ha 0=maradék 360*:n :f [eredmény :n*360/:f]
  eredmény keres :f :n+1
vége

```

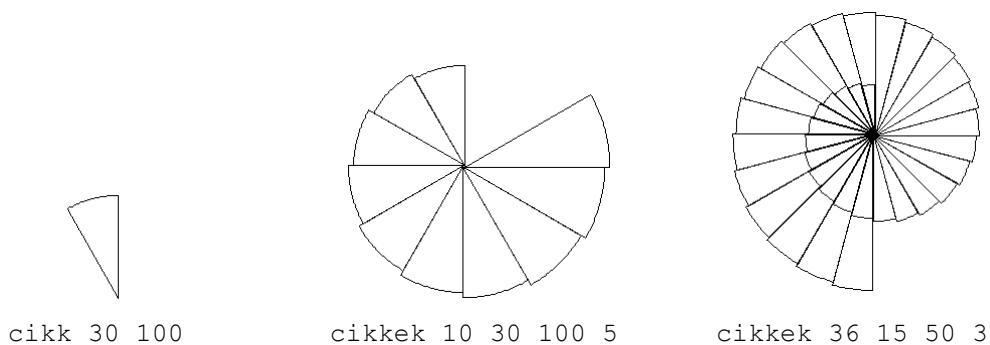
## Rekurzív forgatás, eltolás

A következőkben egy ábrát forgatunk (vagy eltolunk) és közben változtatjuk a méretét. Így különböző alakzatokhoz – pl. spirálokhoz – hasonló ábrákat is kaphatunk.

### Körcikkekből spirál

Készíts Logo eljárást (cikk :szög :sugár), ami egy körcikket tud rajzolni, adott szög és sugár esetén!

Készíts eljárást (cikkek :db :szög :sugár :növekmény), amely az előbbit felhasználva az alábbi ábrákat képes rajzolni:



```

eljárás cikk :szög :sugár
  előre :sugár balra 90,5
  ismétlés :szög [előre 2*:sugár*3,14159/360 balra 1]
  balra 90-0,5 előre :sugár jobbra 180+:szög
vége

```

Itt a rekurzív hívásban a körcikk sugara növekszik.

```

eljárás cikkek :db :szög :sugár :növekmény
  cikk :szög :sugár
  ha :db>1 [balra :szög cikkek :db-1 :szög :sugár+:növekmény :növekmény jobbra :szög]
vége

```

### Legyező

Készíts Logo eljárást (cikk :szög :sugár), ami egy körcikket tud rajzolni, adott szög és sugár esetén!

Készíts eljárást (legyező :db :szög :sugár :növekmény), amely az előbbit felhasználva az alábbi ábrát képes rajzolni!



cikk 30 100



legyező 7 30 150 10

Ebben az esetben az összes körcikk ugyanott kezdődik.

```
eljárás cikk :szög :sugár
  előre :sugár balra 90,5
  ismétlés :szög [előre 2* :sugár*3,14159/360 balra 1]
  balra 90-0,5 előre :sugár jobbra 180+:szög
vége
```

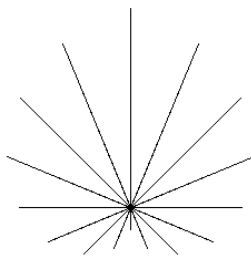
A sugár a rekurzív hívásban csökken, a szög pedig növekszik.

```
eljárás legyező :db :szög :sugár :növ
  cikk :szög :sugár
  ha :db>1 [legyező :db-1 :szög+:növ :sugár-:növ :növ]
vége
```

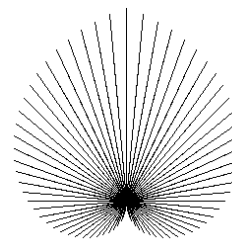
### Pálcikák

Egy játékban adott hosszúságú pálcikákat kell egymáshoz illeszteni a mellékelt ábrának megfelelően.

Készíts Logo eljárást (pálcika :méret :db), amely ilyen ábrát tud készíteni! Az eljárás első paramétere a pálcikák hossza, a második pedig a pálcikák száma legyen! Az egyes pálcikák egymáshoz képest azonos szöggel legyenek elforgatva, az elforgatás helye pedig a pálcika végéhez képest egyenletesen változzon!



pálcika 100 8



pálcika 100 36

A baloldali ábrán a függőleges pálcikát (10,90) arányban osztja a forgatás helye, a következőket (20,80), (30,70), ... arányban.

```
eljárás pálcika :méret :db
  pálcikák :db :méret :méret/(:db+2) 180/:db :méret/(:db+2)
vége
```

```
eljárás pálcikák :db :méret :hol :szög :mennyivel
  hátra :hol előre :méret hátra :méret-:hol balra :szög
  ha :db>1 [pálcikák :db-1 :méret :hol+:mennyivel :szög :mennyivel]
vége
```

## Íves elemek

Készítsd el az íveselem :oldal és az íves :oldal :q :bal :le eljárásokat, az ábrának megfelelően! A :q paraméter legyen a méretcsökkenés aránya, a :bal és a :le pedig a két irányú eltolás aránya a következő íves elem rajzolásához!



```

íveselem 100      íves 100 0,8 0 0      íves 100 0,8 0,8 0,8      íves 100 0,8 -0,8 -0,8
eljárás íves :r :q :bal :le
  ha :r>1 [íveselem :r :bal :le íves :r*:q :q :bal :le]
vége

```

Az íves elem lekerekített sarkú négyzet, a kerekítés egy-egy negyedkör.

```

eljárás íveselem :a :bal :le
  tollatfel előre :a/2*(1+:le) jobbra 90 előre :a/8*3 *(1+:bal) tollatle
  ismétlés 4 [negyedkör :a/8 előre :a/8*6]
  tollatfel hátra :a/8*3*(1+:bal) balra 90 hátra :a/2*(1+:le) tollatle
vége

eljárás negyedkör :r
  ismétlés 90 [előre 2*:r*3,14159/360 jobbra 1]
vége

```

## Mozaik – rekurzívan

A **mozaik** olyan művészeti technika és annak eredménye, amelynél kicsiny méretű színes üveg-, kő- vagy kavicsdarabokból állítják össze a képet vagy mintázatot (néha más anyagokat is használnak). A mozaikdarabokat cementtel, gipszszel rögzítik, esetleg a még nedves vakolatba nyomják bele.<sup>5</sup>

A Logo programozásban olyan mozaikokkal foglalkozunk, amelyek valamely geometrikus mintákból adott szabályszerűséggel épülnek fel, töltenek ki szabályos alakú területet.

Egyszerű esetekben a megoldás két ciklus – ha minden lépésben ugyanazt az elemet kell kirajzolni.

Előfordulhat azonban, hogy lépésenként (soronként, alapelemenként) más a tennivalónk. Ekkor nincs más – egyszerű – lehetőségünk, mint a rekurzív megoldás elkészítése.

### Forgatott elemek

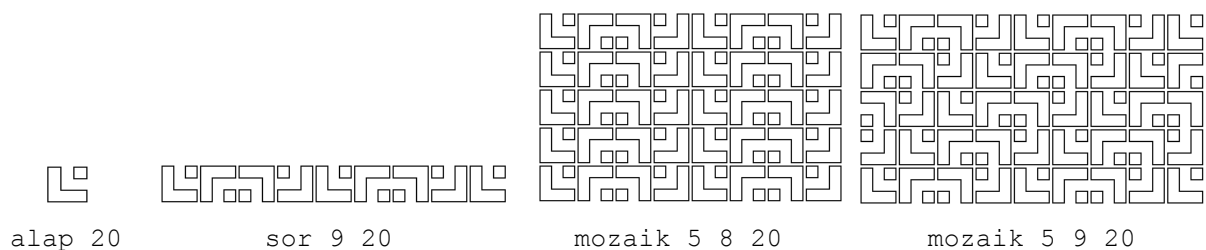
Az egyik megvizsgálandó eset, amikor a mozaik többféle elemből áll, amit nem könnyű ciklikusan megvalósítani.

Ennek a esete, amikor az egyes elemek egymás elforgatottjai.

### Mozaik 90 fokkal forgatott alapelemekkel

Egy mozaik alapeleme az alábbi ábrán látható lemez, amit az alap :hossz eljárással rajzolhatunk meg. A sor :darab :hossz eljárás egymás mellé tesz :darab ilyen lemezt, a szomszédjukhoz képest 90 fokkal elforgatva. A mozaik :sordb :oszlopb :hossz eljárás úgy tesz egymás alá :sordb darab sort, hogy az új sorok elején levő lemez az előző sor végén levő után következő (azaz 90 fokkal elforgatott) legyen. Az egyes lemezek egymástól 3 egység távolságra vannak.

Készítsd el az alap, a sor és a mozaik eljárásokat!



Itt ugyan négyféle elem van, de ezek az alapelem 0, 90, 180 és 270 fokos elforgatottjai. Rekurzív eljárást írunk, amelynek egyik paramétere ez a 4 érték lesz, ciklikusan léptetve. Alapelem rajzolásból azonban elég egyet megírunk, mert az elforgatások csak annyit jelentenek, hogy a négyzet más-más sarkából kell elindulni a rajzolással.

```
eljárás alap :r
  előre :r jobbra 90 előre :r/3 jobbra 90 előre 2*:r/3
  balra 90 előre 2*:r/3 balra 90 tollatfel előre :r/3
  tollatle ismétlés 4 [előre :r/3 balra 90] tollatfel
  hátra :r/3 tollatle jobbra 180 előre :r/3 jobbra 90 előre :r jobbra 90
vége
```

<sup>5</sup> A Wikipédiából, a szabad enciklopédiából

```

eljárás sor :m :r :szög
  forgat :r :szög
  ha :m>1 [tollatfel jobbra 90 előre :r+3 balra 90 tollatle
           sor :m-1 :r maradék :szög+90 360
           tollatfel jobbra 90 hátra :r+3 balra 90 tollatle]
vége

eljárás forgat :r :szög
  tollatfel ismétlés :szög/90 [előre :r jobbra 90]
  tollatle alap :r tollatfel
  ismétlés (360-:szög)/90 [előre :r jobbra 90] tollatle
vége

```

Ezt a mozaikot a változatosság kedvéért a felső sorral kezdjük és az alsóban fejezzük be.

```

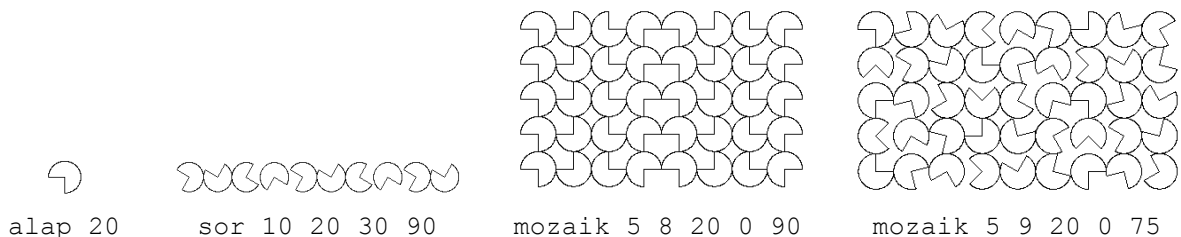
eljárás mozaik :n :m :r :szög
  sor :m :r :szög
  ha :n>1 [tollatfel hátra :r+3 tollatle
           mozaik :n-1 :m :r maradék :szög+90*maradék :m 4 360
           tollatfel előre :r+3 tollatfel]
vége

```

### Mozaik forgatott alapelemekkel

Egy mozaik alapeleme egy háromnegyed körlemez, amit az alap :r eljárással rajzolhatunk meg. A sor :m :r :szög :ford eljárás egymás mellé tesz :m ilyen körlemezt, a szomszédjukhoz képest :ford fokkal elforgatva. A mozaik :n :m :r :szög :ford eljárás úgy tesz egymás alá :n darab sort, hogy az új sorok elején levő körlemez az előző sor végén levő után következő (azaz :ford fokkal elforgatott) legyen!

Készítsd el az alap, a sor és a mozaik eljárásokat!



Ebben az esetben egy körszerű alakzat az alapelem, amit könnyű forgatni tetszőleges szöggel – azt kell kiszámolni, hogy a háromnegyed körívet hol kezdjük rajzolni.

```

eljárás alap :r
  háromnegyed :r jobbra 90 előre :r balra 90 előre :r jobbra 90
vége

eljárás háromnegyed :r
  ismétlés 270 [előre :r/180*3,14159 jobbra 1]
vége

eljárás sor :m :r :szög :ford
  forgat :r :szög
  ha :m>1 [tollatfel jobbra 90 előre 2*:r balra 90 tollatle
           sor :m-1 :r maradék :szög+:ford 360 :ford
           tollatfel jobbra 90 hátra 2*:r balra 90 tollatle]
vége

```



```

eljárás forgat :r :szög
  tollatfel ismétlés :szög [előre :r/180*3,14159 jobbra 1]
  tollatle alap :r tollatfel
  ismétlés 360-:szög [előre :r/180*3,14159 jobbra 1] tollatle
vége

```

Ezt a mozaikot is felülről lefelé rajzoljuk.

```

eljárás mozaik :n :m :r :szög :ford
  sor :m :r :szög :ford
  ha :n>1 [tollatfel hátra 2*:r tollatle
           mozaik :n-1 :m :r maradék :szög+:ford*:m 360
           tollatfel előre 2*:r tollatfel]
vége

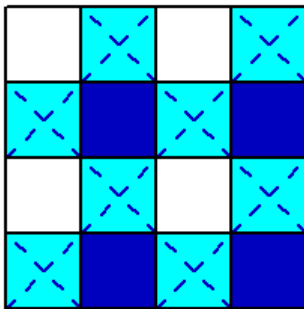
```

## Mozaik többféle elemmel

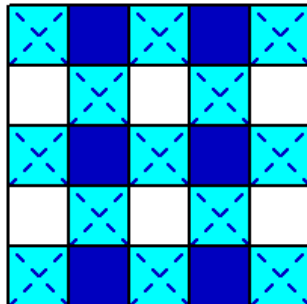
Gyakoribb az az eset, amikor a mozaik kétféle, vagy többféle alapelemből áll, amiket külön-külön meg kell rajzolnunk.

### Kockás abrosz – 1. változat

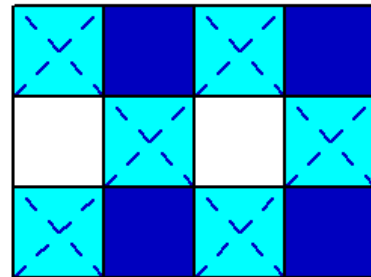
Készítsd el a kockás abroszt (kockásabrosz :sordb :oszlopdb :méret), amely három különböző színű négyzetekből áll és a kockákon keresztül szaggatott vonalak is díszítik az ábrának megfelelően. A :sordb a sorok, az :oszlopdb az oszlopok számát, a :méret pedig a négyzetek méretét jelöli.



kockásabrosz 4 4 50



kockásabrosz 5 5 50



kockásabrosz 3 4 60

Egy logikai értékű paraméter használunk arra, hogy szaggatott vonalas vagy vonal nélküli alapelemet kell rajzolni.

A terítő sorai is kétfélék, azaz két sor rajzoló eljárást kell írni, szintén logikai paraméterrel.

A feladat kiírásban a logikai paraméter nem szerepel, ezért egy ún. fedőeljárást írunk (kockásabrosz), ami meghívja eggyel több paraméterrel a mozaik rajzolót. Ha :lo=igaz, akkor a szaggatott vonalas elemet kell rajzolni, különben pedig a másikat.

```

eljárás kockásabrosz :sordb :oszlopdb :méret :lo
  mozaik :sordb1 :oszlopdb :méret "igaz"
vége

eljárás mozaik :sordb :oszlopdb :méret :lo
  ha :sordb>0 [hakülönbem :lo [sor13 :oszlopdb :méret :lo]
              [sor23 :oszlopdb :méret :lo]
              előre :méret
              mozaik :sordb-1 :oszlopdb :méret nem :lo
              hátra :méret]
vége

```

A terítő sorai nem egyformák, ez az egyik fajta sor.

```
eljárás sor13 :db :méret :lo
  ha :db>0
    [hakülönben :lo [kocka :méret 15 "hamis][kocka :méret 11 "igaz]
      jobbra 90 előre :méret balra 90
      sor13 :db-1 :méret nem :lo
      jobbra 90 hátra :méret balra 90]
  vége
```

A terítő másik fajta sora.

```
eljárás sor23 :db :méret :lo
  ha :db>0
    [hakülönben :lo [kocka :méret 1 "hamis][kocka :méret 11 "igaz]
      jobbra 90 előre :méret balra 90
      sor23 :db-1 :méret nem :lo
      jobbra 90 hátra :méret balra 90]
  vége
```

Egy kocka kirajzolásában a :csík paraméter határozza meg, hogy kell-e szaggatott vonalakból csíkokat is rajzolni bele.

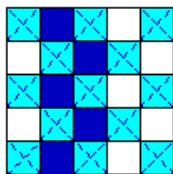
```
eljárás kocka :méret :szín :csík
  töltőszín! :szín ismétlés 4 [előre :méret jobbra 90]
  tollatfel jobbra 45 előre :méret tölt hátra :méret balra 45 tollatle
  ha :csík [jobbra 45 csík :méret*gyök 2 tollatfel
    hátra :méret*gyök 2 jobbra 45 előre :méret
    balra 135 tollatle csík :méret*gyök 2 tollatfel
    hátra :méret*gyök 2 jobbra 135 hátra :méret
    balra 90 tollatle]
  vége
```

Szaggatott vonalat rajzol.

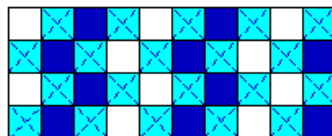
```
eljárás csík :méret
  tollszín! 1 tollatle
  ismétlés 4 [előre :méret/8 tollatfel előre :méret/8 tollatle]
  tollszín! 0
  vége
```

### Kockás abrosz – 2. változat

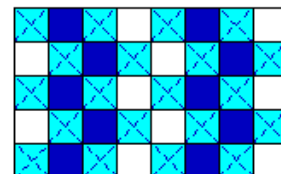
Készítsd el a kockás abroszt (kockásabrosz :sordb :oszlopdb :méret), amely három különböző színű négyzetből áll és a kockákon keresztül szaggatott vonalak is díszítik az ábrának megfelelően. A :sordb a sorok, az :oszlopdb az oszlopok számát, a :méret pedig a négyzetek méretét jelöli.



kockásabrosz 5 5 20



kockásabrosz 4 10 20



kockásabrosz 5 8 20

Itt a sorok négyféle elemet tartalmaznak, kódoljuk ezt négyféle számmal: világoskék – 0 és 2; sötétkék – 1, fehér – 3. A páratlan sorok 0-s, a párosak 3-as kódú elemmel kezdődnek.

```
eljárás kockásabrosz :sordb :oszlopdb :méret
  mozaik :sordb :oszlopdb :méret "igaz
  vége
```

```

eljárás mozaik :sordb :oszlopdb :méret :lo
  ha :sordb>0 [hakülönben :lo [sor :oszlopdb :méret 0]
                [sor :oszlopdb :méret 3]
                előre :méret
                mozaik :sordb-1 :oszlopdb :méret nem :lo
                hátra :méret]
vége

```

A terítő sorai egyformák, csak a mintázat máshol kezdődik. A négyféle elemkód között ciklikusan léptetünk egy paramétert.

```

eljárás sor :db :méret :kezd
  ha :db>0
    [ha :kezd=3 [kocka :méret 15 "hamis]
      ha vagy :kezd=0 :kezd=2 [kocka :méret 11 "igaz]
      ha :kezd=1 [kocka :méret 1 "hamis]
      jobbra 90 előre :méret balra 90
      sor :db-1 :méret maradék :kezd+1 4
      jobbra 90 hátra :méret balra 90]
vége

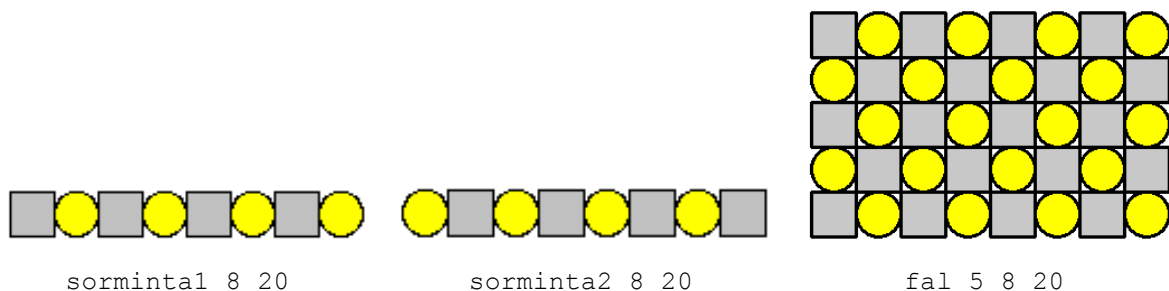
```

Egy kocka kirajzolása ugyanaz, mint az előző feladatban.

### Mozaik kétféle elemből közvetett rekurzióval

Egy térelválasztó falat (ún. spanyolfalat) kétféle elemből építenek össze.

Készíts Logo eljárást (fal :n :m :h), amely ilyen falat tud rajzolni! Legyen :n a sorok, :m az oszlopok száma, :h pedig a négyzet oldalhossza (illetve a kör átmérője)!



A fal eljárásból kettőt készítünk, amelyek egymást hívják – közvetett rekurzió. Az egyik esetén a mozaik bal alsó sarka négyzet, a másikon pedig kör.

```

eljárás fal :m :n :x
  ha :m>0 [sorminta1 :n :x felfelé :x fal2 :m-1 :n :x felfelé -:x]
vége

```

```

eljárás fal2 :m :n :x
  ha :m>0 [sorminta2 :n :x felfelé :x fal :m-1 :n :x felfelé -:x]
vége

```

Alkalmazzunk itt is közvetett rekurziót! Az egyik sorminta négyzettel kezdődő legyen, a másik pedig körrel kezdődő!

```

eljárás sorminta1 :n :x
  ha :n>0 [alapelem1 :x elmozdulás :x sorminta2 :n-1 :x elmozdulás -:x]
vége

```

```

eljárás sorminta2 :n :x
  ha :n>0 [alapelem2 :x elmozdulás :x sorminta1 :n-1 :x elmozdulás -:x]
vége

```

```
eljárás alapelem1 :x
  ismétlés 4 [előre :x jobbra 90]
  töltőszín! 7
  tollatfel jobbra 45 előre :x tölt hátra :x balra 45 tollatle
vége

eljárás alapelem2 :x
  tollatfel előre :x/2 töltőszín! 14 tollatle
  ismétlés 360 [előre :x*3,14159/360 jobbra 1] tollatfel
  jobbra 90 előre :x/2 tölt hátra :x/2 balra 90 hátra :x/2 tollatle
vége

eljárás felfelé :x
  tollatfel előre :x tollatle
vége

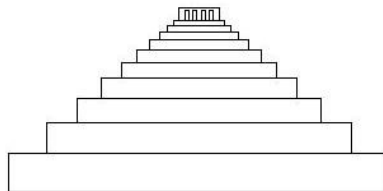
eljárás elmozdulás :x
  tollatfel jobbra 90 előre :x balra 90 tollatle
vége
```

## Különböző méretű elemek

A rekurzió másfajta alkalmazására akkor van szükség, amikor a mozaik elemei különböző méretűek, elhelyezésűek. A méretek valamilyen szabályszerűség szerint változnak.

## Maja templom

Készíts Logo eljárást (maja :n :szél :mag), amely megrajzol egy maja templomot! Az :n az emeletek számát, a :szél a templom alsó szintjének szélességét, a :mag pedig az alsó szint magasságát jelenti. A templom szintjeinek szélessége és magassága arányosan csökken. Az épület tetején egy 4 ajtós építmény látható.



maja 10 100 30

A feladatszövegben nem rögzített, ezért megbecsüljük a méretcsökkenést: legyen pl. 80%-os! A méret változása itt nemcsak a téglarajzolást befolyásolja, hanem az egyes téglák elhelyezését is egymáshoz képest.

```
eljárás maja :n :szél :mag
  hakülönben :n>0 [tégla :szél :mag következő :szél :mag
    maja :n-1 :szél*0,8 :mag*0,8 előző :szél :mag]
    [felépítmény :szél :mag]
vége

eljárás következő :szél :mag
  előre :mag jobbra 90 előre :szél/10 balra 90
vége

eljárás előző :szél :mag
  jobbra 90 hátra :szél/10 balra 90 hátra :mag
vége
```

Téglalap, amelyből a lépcsősor is áll.

```
eljárás téglalap :szél :mag
  ismétlés 2 [előre :mag jobbra 90 előre :szél jobbra 90]
vége
```

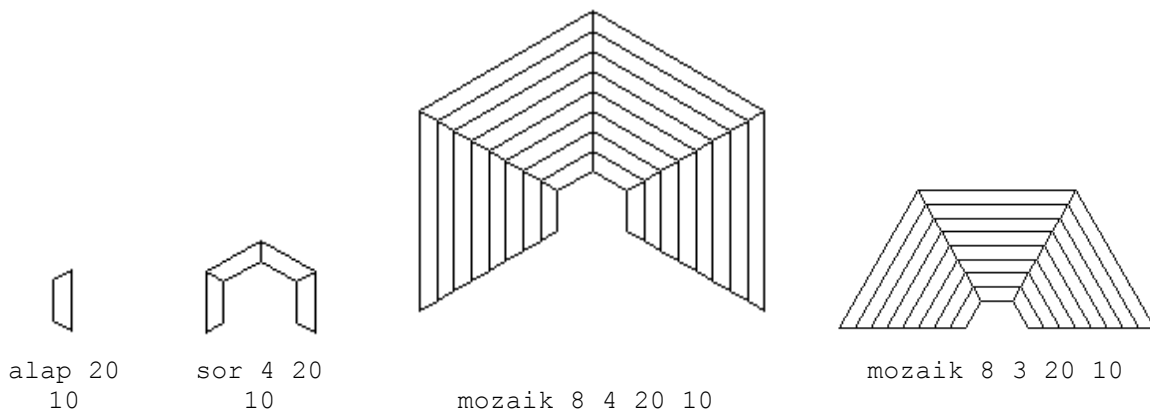
A lépcsők tetején elhelyezkedő épület.

```
eljárás felépítmény :szél :mag
  téglalap :szél :mag*3 jobbra 90 előre 0,15*:szél balra 90
  ismétlés 4 [téglalap :szél*0,1 :mag*2,5 jobbra 90
    előre :szél*0,2 balra 90]
vége
```

### Színház

Szabadtéri színházak nézőterét sokszor egy sokszög oldalai mentén, egyre hosszabb sorokkal rendezik be, középre helyezve a színpadot.

Készíts Logo programot, amely az ábrán látható alapelemet tudja rajzolni (alap :a :b, ahol :a a trapéz párhuzamos oldalai közül a rövidebb hossza, :b pedig az oldalának a hossza)! Az alapelemeket egymás mellé rakja egy hatszög :n oldalára (sor :n :a :b), ahol :n legfeljebb 6. A harmadik egymás köré rakva egy területmintát alakít ki (mozaik :m :n :a :b) :m sorból!



```
eljárás alap :méret :oldal
  előre :méret jobbra 60 előre :oldal jobbra 120
  előre :oldal+:méret jobbra 120 előre :oldal jobbra 60
vége
```

```
eljárás sor :n :méret :oldal
  ismétlés :n [alap :méret :oldal előre :méret balra 60]
  ismétlés :n [jobbra 60 hátra :méret]
vége
```

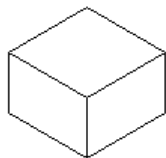
```
eljárás mozaik :m :n :méret :oldal
  sor :n :méret :oldal
  ha :m>1 [jobbra 120 előre :oldal balra 120
    mozaik :m-1 :n :oldal+:méret :oldal]
vége
```

## Sorok hossza különböző

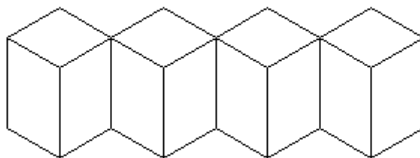
A következő feladatokban az alapelemek egyformák a teljes mozaikban. Most az alapelemek soronkénti száma fog változni. Emiatt a sorok rajzolása történhet ciklussal, de a sorokat hívó mozaiknak rekurzívnak kell lenni.

### Piramis

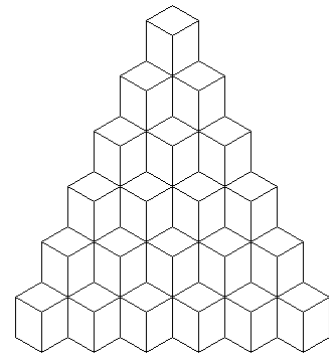
Egy szabályos hatszög két párhuzamos oldalának hosszát megváltoztattuk, majd a belsejébe az ábrának megfelelően 3 vonalat húztunk (tégla :a :b). A téglákat egymás mellé tehetjük (sor :db :a :b), illetve piramist építhetünk belőle (piramis :db :a :b). A piramis sorokból áll, amelyek felfelé haladva egyre kevesebb téglából állnak.



tégla 20 10



sor 4 10 15



piramis 6 5 8

```

eljárás tégla :a :b
  előre :b jobbra 60
  ismétlés 3 [előre :a jobbra 60 előre :a jobbra 120]
  balra 60 előre :b jobbra 60 előre :a jobbra 120 előre :b
  balra 60 előre :a jobbra 60
vége
  
```

A sor elejére kell visszatérni, amit az alaplap élei mentén teszünk meg.

```

eljárás sor :db :a :b
  ismétlés :db [tégla :a :b jobbra 120 előre :a balra 60
                előre :a balra 60]
  ismétlés :db [balra 120 előre :a jobbra 60 előre :a jobbra 60]
vége
  
```

A sorok egyre rövidülnek, így ekkor rekurzióval érdemes dolgozni. A következő sor elejére az élek mentén jutunk.

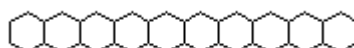
```

eljárás piramis :db :a :b
  ha :db>0 [sor :db :a :b előre :b jobbra 60 előre :a balra 60
            piramis :db-1 :a :b jobbra 60 hátra :a balra 60 hátra :a]
vége
  
```

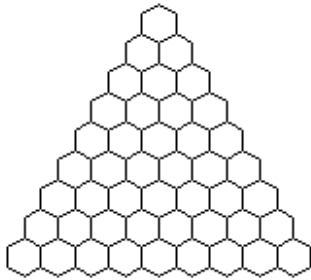
### Hatszögmozaikok

Szabályos hatszöget is vehetünk a mozaik alapjául. Hatszögekkel a sík sokféle tartománya fedhető le, ezek közül nézünk meg néhányat a továbbiakban.

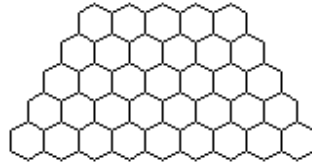
Készíts Logo eljárást, amely méhsejtből (szabályos hatszögekből) különböző alakzatokat tud építeni! A hatszög :méret eljárás egyetlen méhsejtet rajzoljon, ahol :méret a hatszög oldalhossza! A sor :n :méret eljárás :n darab méhsejtet rajzoljon egymás mellé:



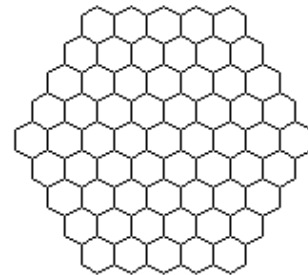
A három `:n :méret`, a trapéz `:n :m :méret` és a hat `:n :méret` eljárások pedig az alábbi ábrákat rajzolják, ahol `:n` az alsó sorban levő hatszögek száma, `:méret` pedig a hatszögek oldalhossza. A trapéznál `:m` a felső sorban levő hatszögek száma legyen!



három 9 20



trapéz 9 5 20



hat 5 20

Ha a hatszög nem állapotátlátszó, hanem a baloldali alsó csúcsából kezdjük rajzolni és a jobboldali alsó csúcsában fejezzük be, akkor a sor rajzolása sokkal egyszerűbb lesz.

```
eljárás hatszög :hossz
  ismétlés 10 [előre :hossz jobbra 60] jobbra 120
vége
```

```
eljárás sor :n :hossz
  ismétlés :n [hatszög :hossz]
  ismétlés :n [balra 120 ismétlés 2 [előre :hossz jobbra 60]]
vége
```

A háromszög lefedése egyszerű, ugyanannyi sort kell rajzolni, ahány elem az alsó sorban van.

```
eljárás három :n :hossz
  sor :n :hossz
  ha :n>1 [előre :hossz jobbra 60 előre :hossz balra 60
           három :n-1 :hossz
           jobbra 60 hátra :hossz balra 60 hátra :hossz]
vége
```

A trapéz csak kicsit bonyolultabb, a rekurzió feltételét kell módosítani!

```
eljárás trapéz :n :m :hossz
  sor :n :hossz
  ha :n>:=m [előre :hossz jobbra 60 előre :hossz balra 60
            trapéz :n-1 :m :hossz
            jobbra 60 hátra :hossz balra 60 hátra :hossz]
vége
```

Megjegyzés: `három :n :hossz ≡ trapéz :n 1 :hossz`

A hatszög lefedés lehet két trapéz lefedés, köztük a megfelelő elmozdulással.

```
eljárás hat :n :hossz
  trapéz :n*2-1 :n :hossz
  ismétlés :n*2-1 [ismétlés 2 [balra 60 hátra :hossz] jobbra 120]
  előre :hossz trapéz :n*2-1 :n (-:hossz)
vége
```

## Optikai csalódások

Egy optikai csalódást (más néven vizuális illúziót) vizuálisan észlelt képek jellemeznek, melyek eltérnek az objektív valóságtól. A szem által gyűjtött információ az agyban kerül feldolgozásra, hogy egy olyan észleletet adjon, ami nem egyezik az ingerforrás egy fizikai mértékével. Három fő típust különböztetünk meg:

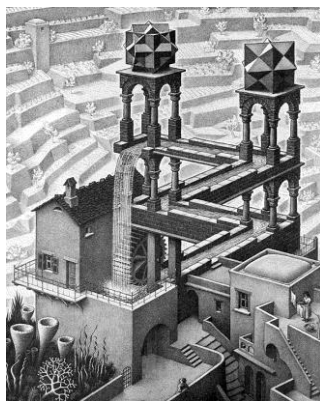
- szó szerinti optikai csalódásokat, melyek olyan képeket hoznak létre, melyek különböznek azoktól a tárgytól, amelyek kiváltják őket;
- fiziológiai illúziókat, melyek hatást gyakorolnak a szemek és az agy egy jellegzetes típusának túlzott ingerlésére (világosság, dőlés, szín, mozgás);
- és kognitív illúziókat, ahol a szem és az agy tudattalan következtetéseket von le. Ezek „elme játékokként” is ismertek lehetnek. (Wikipédia)

### Néhány klasszikus példa:

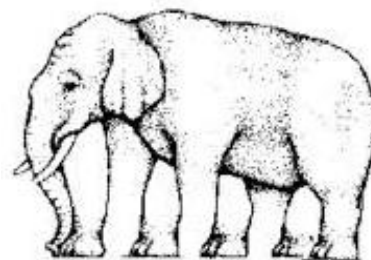
E. G. Boring, amerikai pszichológus képe a legismertebb kétértelmű ábrák egyike. Fiatal hölgyként profilból látjuk, míg idős nőként az állból nagy orr lesz.



M.C.Escher a lehetetlen háromszög konstrukciót felhasználta Vizesés című litográfiájában.



Végül egy újabb klasszikus: Hány lába van az elefántnak? És hol is vannak?



Logo programozásban optikai csalódásokat láthatunk sokszögeknél, sormintáknál, de leggyakrabban mozaikoknál.

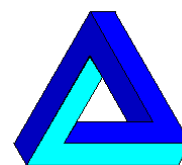
### Térbeli (paradox) illúziók

A paradox illúziókat olyan tárgyak váltják ki, melyek paradoxak vagy lehetetlenek, mint például a Penrose-háromszögben vagy a lehetetlen lépcsősorokban láthatjuk.

Sok olyan síkbeli ábra van, amely látszólag egy térbeli testet ábrázol, azonban ilyen test nem létezhet. Egy ilyenről szól a következő feladat.

### Penrose háromszög

Rajzold meg háromféle kék színnel a mellékelt háromszöget (háromszög)! (Segítség: egy 80 oldalhosszúságú háromszögnek levágtunk a három sarkából egy-egy 10 oldalhosszúságú háromszöget, majd 10 egység széles sávokat színeztük.)





A háromszög különböző stílusú oldalakból áll.

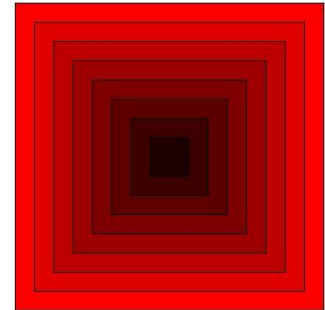
```
eljárás háromszög :h
  jobbra 30
  oldal :h 1 tollatfel előre :h jobbra 120 tollatle
  oldal :h 9 tollatfel előre :h jobbra 120 tollatle
  oldal :h 11 tollatfel előre :h jobbra 120 tollatle
  balra 30
vége

eljárás oldal :h :szín
  tollatfel előre :h/8 tollatle
  előre 6*:h/8 jobbra 120 előre 5*:h/8 jobbra 120
  előre :h/8 jobbra 60 előre 3*:h/8 balra 120
  előre 5*:h/8 jobbra 120 előre :h/8
  tollatfel jobbra 90 előre :h/16 töltőszín! :szín tölt
  hátra :h/16 balra 90 jobbra 60 hátra :h/8 tollatle
vége
```

Josef Albert német festőművész művei inspirálta feladat: a színárnyalatok térbeli képzetet adnak az ábrának:

### Térhatás színekkel

Készítsd el a minta szerinti ábrát (rács :db :hossz)! A belső négyzet oldala (:hossz\*2) egység hosszú, a sávok :hossz vastagságúak. Az egyes sávokat kifelé haladva egyre világosabb pirosra színezd!



Egy :red piros árnyalatú töltőszínt az alábbi utasítással választhatsz: töltőszín! (lista :red 0 0).

A rekurzív hívás miatt szükség van egy plusz paraméterre a növekvő négyzetek oldalának változásának mértékére is :d. Indulásnál ez fele akkora legyen, mint az :oldal értéke! A :db értéke csak akkora lehet, hogy a szín piros árnyalata 0 és 255 közé essen!

```
eljárás rács :db :oldal :d
  ha :db>0 [négyzet :oldal tollatfel jobbra 45 előre :d/2*gyök 2
    töltőszín! (lista 275-:db*25 0 0) tölt
    hátra :d*3/2*gyök 2 balra 45 tollatle
    rács :db-1 :oldal+2*:d :d]
vége
```

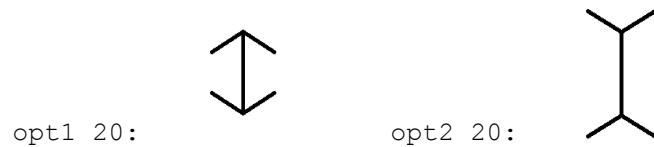
```
eljárás négyzet :oldal
  ismétlés 4 [e :oldal jobbra 90]
vége
```

### Torzító illúziók

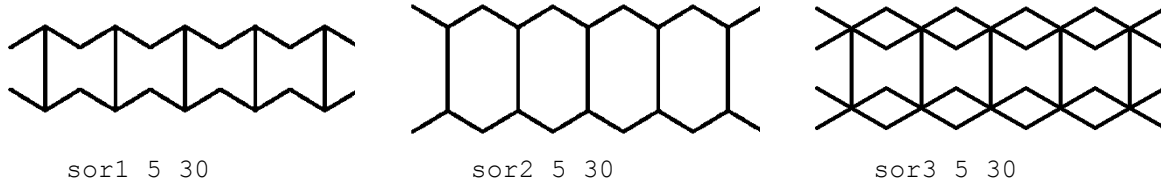
Sok esetben válunk optikai csalódás áldozatává, amikor egy ábrát másnak látunk, mint amilyen a valóságban, mint a következő feladatban. A torzító illúziókat a méret, a hossz vagy a görbület torzulásai jellemzik.

### Müller-Lyer-illúzió

Rajzold meg a két mintát (opt1 :hossz, opt2 :hossz): a függőleges vonalak :hossz\*2 egység hosszúak, a ferde vonalak :hossz egység hosszúak legyenek! A bezárt szögek 60, illetve 120 fokok legyenek!



A két mintaelemenben a függőleges egyenesek különböző hosszúaknak látszanak. Ez azonban nem igaz. Ennek igazolására készíts a mintákból sorokat (sor1 :darab :hossz, sor2 :darab :hossz, sor3 :darab :hossz)! A harmadik igazolja, hogy a függőlegesek azonos hosszúak. A :darab paraméter mutatja meg, hogy a mintaelemekből hány darabot használtunk fel.



```
eljárás opt1 :hossz
  balra 60 vonal :hossz jobbra 120 vonal :hossz balra 60 előre :hossz*2
  balra 120 vonal :hossz balra 120 vonal :hossz balra 120 hátra :hossz*2
vége
```

```
eljárás opt2 :hossz
  balra 120 vonal :hossz balra 120 vonal :hossz balra 120 előre :hossz*2
  balra 60 vonal :hossz jobbra 120 vonal :hossz balra 60 hátra :hossz*2
vége
```

```
eljárás vonal :hossz
  előre :hossz hátra :hossz
vége
```

```
eljárás sor1 :n :hossz
  ismétlés :n [opt1 :hossz jobbra 60 előre :hossz jobbra 60 előre :hossz
               balra 120]
  tollatfel
  ismétlés :n [jobbra 120 hátra :hossz balra 60 hátra :hossz balra 60]
  tollatle
vége
```

```
eljárás sor2 :n :hossz
  ismétlés :n [opt2 :hossz jobbra 120 előre :hossz balra 60 előre :hossz
               balra 60]
  tollatfel
  ismétlés :n [jobbra 60 hátra :hossz jobbra 60 hátra :hossz balra 120]
  tollatle
vége
```

```
eljárás sor3 :n :hossz
  sor1 :n :hossz sor2 :n :hossz
vége
```

A torzító illúziók másik csoportjában nem egyenesnek látjuk az egyenes vonalakat.

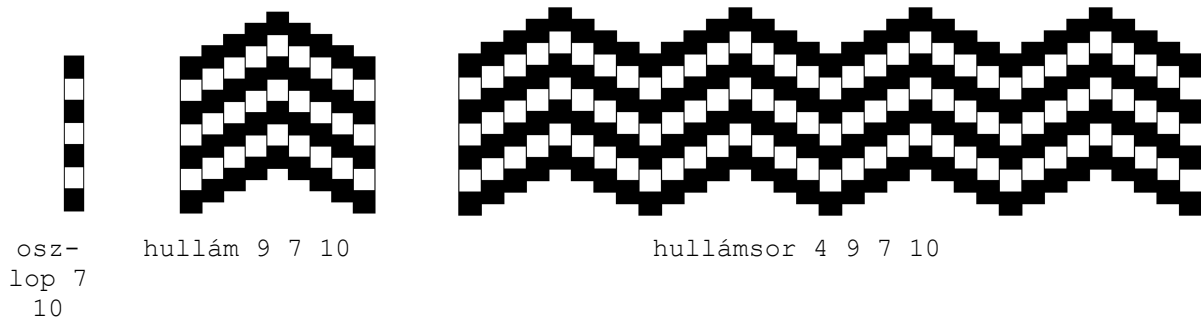
### Kávéház fal illúzió

Készíts mozaik-rajzoló eljárást (oszlop, hullám, hullámsor), amely egymástól négyzetoldalni egységnyire lévő négyzetekből épül fel! A szomszédos négyzetek fél négyzetoldallal vannak eltolva, a hullámsorok egymástól egy négyzetoldalni távolságban vannak!

Az oszlop :n :h eljárás :n darab fekete és fehér, :h oldalhosszúságú négyzetekből állítson össze egy oszlopot (feltehetjük, hogy :n páratlan)! A hullám :m :n :h eljárás :m oszlopot

tegyen egymás mellé az ábrának megfelelően (feltehető, hogy  $m$  is páratlan)! A hullámsor  $:db$   
 $:m$   $:n$   $:h$  eljárás  $:db$  darab hullámot tegyen egymás mellé az ábrának megfelelően!

Példa:



Minden oszlopban üres és teli elemek váltakoznak felváltva:

```
eljárás oszlop :n :h
  ismétlés :n/2 [telielem :h előre :h üreselem :h előre :h]
  telielem :h hátra (:n-1)*:h
vége
```

```
eljárás telielem :h
  üreselem :h
  tollatfel jobbra 45 előre :h tölt hátra :h balra 45 tollatle
vége
```

```
eljárás üreselem :h
  ismétlés 4 [előre :h jobbra 90]
vége
```

Egy hullám  $:m/2$  felfelé mozduló, majd  $:m/2$  lefelé mozduló oszlopból áll, plusz egy utolsóból:

```
eljárás hullám :m :n :h
  ismétlés :m/2 [oszlop :n :h jobbra 90 előre :h balra 90 előre :h/2]
  ismétlés :m/2 [oszlop :n :h jobbra 90 előre :h balra 90 hátra :h/2]
  oszlop :n :h
vége
```

```
eljárás hullámok :db :m :n :h
  ismétlés :db [hullám :m :n :h]
vége
```

### Görbe oldalú négyzet

A következő görbülőnek látszó egyenes már egyetlen ábrán is látszik, de látványosabb egy mozaikon.

Írj Logo eljárásokat (körök, négyzetes, sor, mozaik), amelyek az alábbi ábrákat rajzolják!

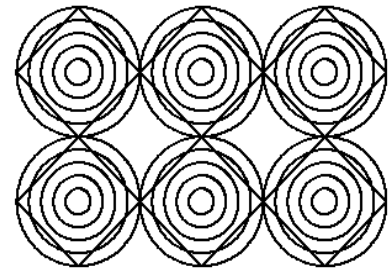
A körök  $:db$   $:r$   $:kül$  eljárás  $:db$  darab koncentrikus kört rajzoljon, a legkisebb sugara legyen  $:r$  és a sugarak különbsége is legyen  $:kül=:r$ ! A négyzetes  $:db$   $:r$  eljárás a legnagyobb körhöz egy négyzetet illesszen az ábrának megfelelően! A sor  $:s$   $:db$   $:r$  eljárás  $:s$  darab ilyen ábrát tegyen egymás mellé! A mozaik  $:o$   $:s$   $:db$   $:r$  eljárás pedig  $:o$  darab sort tegyen egymás fölé!



négyzetes 5 10



sor 3 5 10



mozaik 2 3 5 10

Ha a köröket a középpontjukból kezdjük rajzolni, akkor könnyű lesz a koncentrikus körök rajzolása:

```
eljárás kkör :r
  tollatfel előre :r jobbra 90 tollatle
  ismétlés 360 [előre :r*3,14159/180 jobbra 1]
  tollatfel balra 90 hátra :r tollatle
vége
```

```
eljárás körök :db :h :kül
  kkör :h
  ha :db>1 [körök :db-1 :h+:kül :kül]
vége
```

A legkülső körbe rajzolhatunk egy négyzetet, az egyetlen nehéz – matematikai – feladat a négyzet oldalhosszának kiszámítása:

```
eljárás négyzetes :db :h
  körök :db :h :h
  tollatfel előre :h*:db jobbra 135 tollatle
  ismétlés 4 [előre :h*:db*gyök 2 jobbra 90]
  tollatfel balra 135 hátra :h*:db tollatle
vége
```

Ezután a kapott elemből mozaikot már a szokásos módon gyárthatunk:

```
eljárás sor :s :db :r
  ismétlés :s [négyzetes :db :r tollatfel jobbra 90
    előre 2*:r*:db balra 90 tollatle]
  tollatfel jobbra 90 hátra :s*2*:r*:db balra 90 tollatle
vége

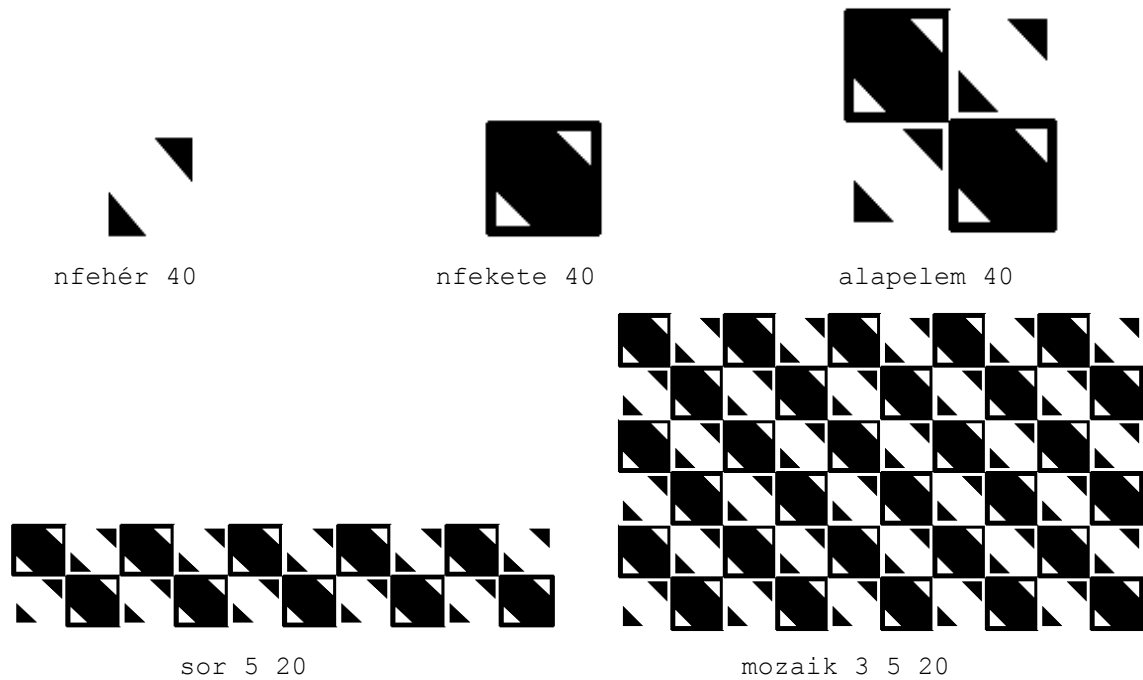
eljárás mozaik :o :s :db :r
  ismétlés :o [sor :s :db :r tollatfel előre 2*:r*:db tollatle]
  tollatfel hátra :o*2*:r*:db tollatle
vége
```

### Ferde csíkok

Amíg az előző ábrán a körök és egyenesek viszonya miatt láttunk görbe vonalakat, addig a következő a fekete-fehér folthatások miatt olyan, mintha balra dőlné.

Készíts Logo eljárásokat az alábbi mozaik rajzolására! Az ábra fekete sarkú fehér, illetve fehér sarkú fekete négyzetekből áll. *Ha másképp látod, az csak érzékszalódás!*

Az nfehér :h és az nfe fekete :h eljárás rajzolja a :h oldalhosszú fehér, illetve fekete négyzetet! Az alapelem :h eljárás 2-2 ilyen négyzetet építsen össze! A sor :m :h eljárás egy sorba :m alapelemet rajzoljon! A mozaik :n :m :h eljárás pedig :n sort rajzoljon egymás fölé!



A fehér négyzet két fekete derékszögű háromszögből (sarok) és fehér határvonalból áll.

```

eljárás nfehér :h
  ismétlés 2 [sarok :h/2 tollatfel ismétlés 2 [előre :h jobbra 90] tollatle]
  tollvastagság! 5 tollszín! 15
  ismétlés 4 [előre :h jobbra 90]
  tollszín! 0
vége
  
```

A fekete négyzet egy fekete (nem szabályos) hatszögből és fekete határvonalból áll.

```

eljárás nfekete :h
  előre :h/2
  ismétlés 2 [előre :h/2 jobbra 90 előre :h/2 jobbra 45
    előre :h/2*gyök 2 jobbra 45]
  tollatfel jobbra 45 előre :h/2 tölt hátra :h/2 balra 45
  hátra :h/2 tollatle tollvastagság! 5
  ismétlés 4 [előre :h jobbra 90]
  tollvastagság! 1
vége
  
```

A sarok fekete derékszögű háromszög.

```

eljárás sarok :h
  előre :h jobbra 135 előre :h*gyök 2 jobbra 135 előre :h jobbra 90
  tollatfel jobbra 45 előre :h/2 tölt hátra :h/2 balra 45 tollatle
vége
  
```

Az alap két szemközti sarokból megrajzolt fehér-fekete négyzetpárt tartalmaz.

```

eljárás alap :h
  ismétlés 2 [nfehér :h tollatfel előre :h+3 tollatle nfekete :h
    tollatfel előre :h jobbra 90 előre :h*2+3 jobbra 90 tollatle]
vége
  
```

A sor és a mozaik rajzolása ezek után a klasszikus eljárásokkal megvalósítható.

```
eljárás mozaik :n :m :h
  ismétlés :n [sor :m :h tollatfel előre 2*:h+6 tollatle]
  tollatfel hátra (:h+3)*:n*2 tollatle
vége

eljárás sor :m :h
  ismétlés :m [alap :h tollatfel jobbra 90 előre 2*:h+6
               balra 90 tollatle]
  tollatfel jobbra 90 hátra :m*2*(h+3) balra 90 tollatle
vége
```

### Fekete-fehér illúzió

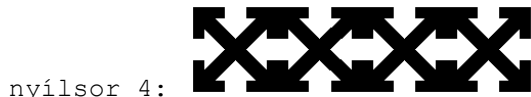
Fekete és fehér színek váltakozása sokszor becsapja szemünket, mint az a következő feladatban látható.

### Nyilak

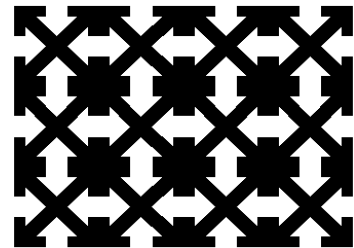
Vajon fekete vagy fehér nyilak vannak rajta? A válasz: mindkettő! A lényeg: olyan elemeket is látunk, amit nem rajzoltunk meg. A fehér nyilak már a 4 fekete nyílból készült ábrán is láthatóak, de igazából a belőlük készült mozaikon feltűnőek.



Rajzolj négy nyilat az ábrának megfelelően (nyilak :h), az ábrán a nyíl szakaszai  $3 \cdot h$ , illetve 10 egység hosszúak legyenek! A nyilakat rendezd sorba (nyílsor :n :h), amelyben :n nyílnégyes van egymás mellett! A sorokat helyezd egymás fölé (nyílminta :m :n :h), amelyben :m sor van egymás fölött!



nyílminta 3 4:



Ha egy nyilat nem állapotátlátszóan írunk meg, akkor a 4 nyílből álló alapábránk egyszerű elforgatásokkal keletkezhethet a nyílből:

```
eljárás nyíl :h
  előre :h/8*3 jobbra 135 ismétlés 2 [előre :h/8 balra 90]
  ismétlés 2 [előre 3*:h/8 balra 90] előre :h/8 balra 90
  előre :h/8 jobbra 135 előre :h/8*3
vége

eljárás nyilak :h
  jobbra 45 ismétlés 4 [nyíl :h jobbra 90]
  balra 45 tollatfel előre 5 tölt hátra 5 tollatle
vége
```

A sor és a mozaik itt is a klasszikus feladatmegoldás:

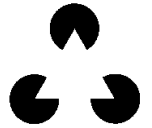
```

eljárás nyílsor :n :h
  ismétlés :n [nyilak :h tollatfel jobbra 45 előre 3*:h/8
               jobbra 45 előre :h/4 jobbra 45 előre 3*:h/8
               balra 45 előre :h/4 balra 90 tollatle]
  tollatfel
  ismétlés :n [balra 90 előre :h/4 jobbra 45 előre 3*:h/8
               balra 45 előre :h/4 balra 45 előre 3*:h/8 jobbra 135]
  tollatle
vége

eljárás nyílminta :m :n :h
  ismétlés :m [nyílsor :n :h tollatfel jobbra 45 előre 3*:h/8
               balra 45 előre :h/2 balra 45 előre 3*:h/8
               jobbra 45 tollatle]
vége
  
```

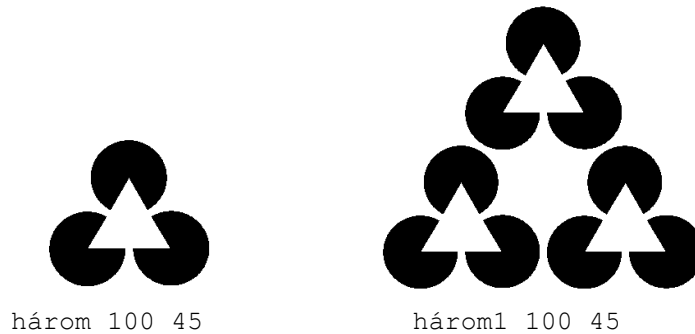
### Körök és háromszögek

Fekete és fehér színek váltakozása sokszor becsapja szemünket, mint az a következő feladatban látható – itt már olyat is „látunk”, ami nincs is ott. Hol van az ábrán a háromszög?



Rajzolj az ábrának megfelelő, szabályos háromszöget tartalmazó körökből (három :h :r), ahol a körök sugara :r, az általuk körbevett fehér háromszög oldala pedig :h hosszúságú.

A kapott ábrát elhelyezhetjük egymás mellett: három1 :h :r!



A megoldásban a háromszög sarkai köré 300 fokos festett körökből rajzolunk.

```

eljárás körív :r :sz
  ismétlés :sz [előre :r*3,14159/180 jobbra 1]
vége

eljárás három :h :r
  ismétlés 3 [tollatfel előre :h tollatle hátra :r balra 90
              körív :r 300 jobbra 90 előre :r tollatfel
              előre :r/2 tölt hátra :r/2 balra 180 tollatle]
vége

eljárás három1 :h :r
  ismétlés 3 [három :h :r tollatfel előre :h*3 jobbra 120 tollatle]
vége
  
```

### Hullámzás

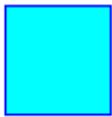
A következő feladatban két, ártatlannak tűnő színminta egymásra rajzolásával érünk el meglepő eredményt. Készíts Logo eljárásokat az alábbi, érzékcsalódást okozó ábrák megrajolására!

A négyzet :h :külső :belső eljárás a :h oldalhosszúságú négyzet oldalát :külső, a belsejét pedig :belső színnel rajzolja ki!

Az alapsor  $:m :h$  eljárás  $:m$  darab zöld belsejű és határvonalú négyzetet rajzoljon egy sorba, az alapmozaik  $:n :m :h$  eljárás  $:n$  sort egymás fölé! A négyzetek közötti távolság legyen egyenlő a négyzet oldalhosszával!

A felsősor  $:m :h :s$  eljárás az ábrának megfelelő mintázatban  $:m$  darab fehér belsejű és piros határvonalú, valamint piros belsejű és fehér határvonalú négyzetet rajzoljon egy sorba, az felsőmozaik  $:n :m :h :s$  eljárás  $:n$  sort egymás fölé! A négyzetek közötti távolság legyen egyenlő a négyzet oldalhossza felével! A  $:s$  az alsó sor első kitöltött négyzetének a sorszáma legyen!

A mozaik  $:n :m :h$  eljárás a mintának megfelelően helyezze el az alapmozaikra a felsőmozaikot! Az alapmozaik négyzetei mérete  $3 * :h$ , a felsőmozaiké pedig  $2 * :h$  legyen!



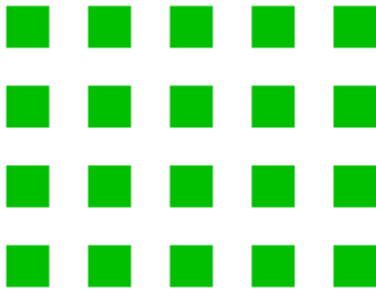
négyzet 60 1 10



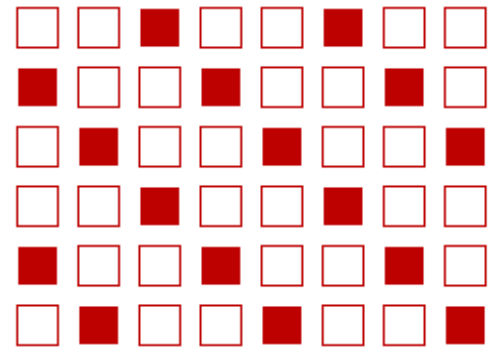
alapsor 5 30



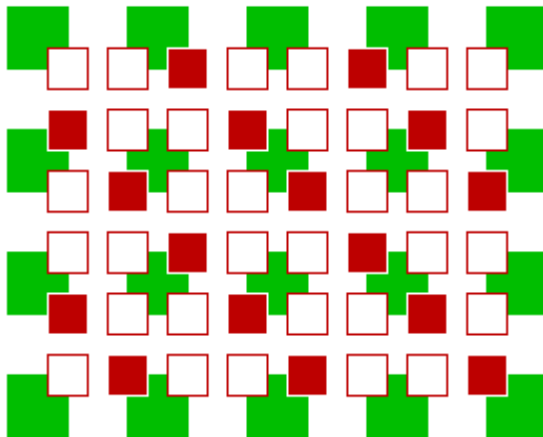
felsősor 8 30 2



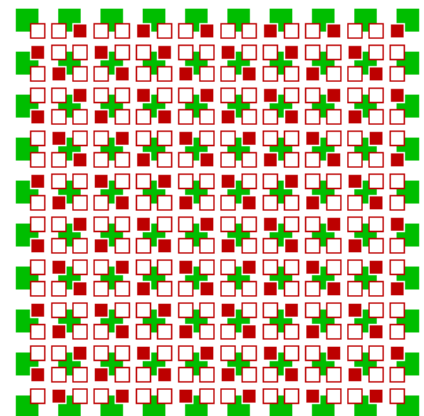
alapmozaik 4 5 30



felsőmozaik 6 8 30 2



mozaik 4 5 60



mozaik 10 10 10

```

eljárás mozaik :n :m :h
  alapmozaik :n :m 3*:h
  tollatfel előre 2*:h jobbra 90 előre 2*:h balra 90 tollatle
  felsőmozaik 2*( :n-1) 2*( :m-1) 2*:h 2
  tollatfel jobbra 90 hátra 2*:h balra 90 hátra 2*:h tollatle
vége
    
```



```

eljárás alapmozaik :n :m :h
  ismétlés :n [alapsor :m :h tollatfel előre 2*:h tollatle]
  tollatfel hátra 2*:n*:h tollatle
vége

eljárás alapsor :m :h
  ismétlés :m [négyzet :h 2 2 tollatfel jobbra 90 előre 2*:h
    balra 90 tollatle]
  tollatfel jobbra 90 hátra 2*:m*:h balra 90 tollatle
vége

eljárás felsőmozaik :n :m :h :i
  felsősor :m :h :i
  ha :n>1 [tollatfel előre 1,5*:h tollatle
    felsőmozaik :n-1 :m :h maradék (:i+1) 3
    tollatfel hátra 1,5*:h tollatle]
vége

eljárás felsősor :m :h :i
  hakülönben :i=0 [négyzet :h 15 4][négyzet :h 4 15]
  ha :m>1 [tollatfel jobbra 90 előre 1,5*:h balra 90 tollatle
    felsősor :m-1 :h maradék (:i+1) 3
    tollatfel jobbra 90 hátra 1,5*:h balra 90 tollatle]
vége

eljárás négyzet :h :s :bs
  tollvastagság! 1 ismétlés 4 [előre :h jobbra 90]
  jobbra 45 tollatfel előre :h töltőszín! :bs töltőmód! 1 tölt
  hátra :h balra 45 tollatle tollszín! :s
  ismétlés 4 [előre :h jobbra 90] tollszín! 0
vége

```

### Körök vagy négyzetek?

A következő feladatban a négyzetek körnek tűnnek.

Az alábbi mozaik kék és zöld négyzetlapokból áll, amelyek sarkában fehér és lila díszítő elemek vannak! *Ha középben köröket látsz, az csak érzékcsalódás!*

A négyzet :h :szín eljárás egy :h oldalhosszúságú, :szín színű négyzetet rajzoljon! A sor :m :h :szín eljárás :m négyzetet tegyen egymás mellé, ahol :szín a sor első négyzetének a színe legyen! A tér :n :m :h eljárás :n sort rajzoljon egymás fölé!

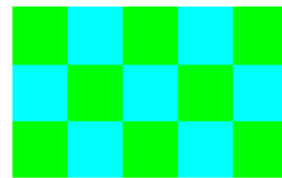
Az ábra :h :szín eljárás egy :szín színű, :h oldalhosszú mintát rajzoljon! A négyzetlapokra az minta :n :m :h eljárás :n sorból és :m oszlopból álló mozaikot rajzol.



négyzet 50  
10



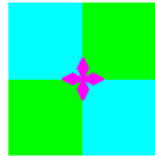
sor 5 40 10



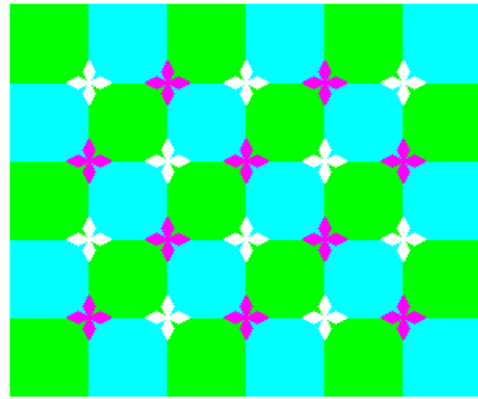
tér 3 5 40



ábra 30 13



mozaik 2 2 50



mozaik 5 6 50

```

eljárás mozaik :n :m :h
  tollatfel tér :n :m :h 10 előre :h jobbra 90 előre :h balra 90
  minta :n-1 :m-1 :h 13 jobbra 90 hátra :h balra 90 hátra :h tollatle
vége

```

A négyzetek felváltva zöld és kék színűek. 21-ből kivonva az egyik színkódját, kapjuk meg a másikat.

```

eljárás tér :n :m :h :s
  sor :m :h :s
  ha :n>1 [előre :h tér :n-1 :m :h 21-:s hátra :h]
vége

eljárás sor :m :h :s
  tollszín! :s töltőszín! :s négyzet :h
  ha :m>1 [jobbra 90 előre :h balra 90 sor :m-1 :h 21-:s
          jobbra 90 hátra :h balra 90]
vége

eljárás négyzet :h
  tollatle ismétlés 4 [előre :h jobbra 90]
  tollatfel jobbra 45 előre :h tölt hátra :h balra 45
vége

```

Az ábrák felváltva lila és fehér színűek, színkódjuk 13 és 15. Emiatt 28-ból kivonva az egyik kódot, kapjuk meg a másikat.

```

eljárás minta :n :m :h :s
  mintasor :m :h :s
  ha :n>1 [előre :h minta :n-1 :m :h 28-:s hátra :h]
vége

eljárás mintasor :m :h :s
  tollszín! :s töltőszín! :s ábra :h
  ha :m>1 [jobbra 90 előre :h balra 90 mintasor :m-1 :h 28-:s
          jobbra 90 hátra :h balra 90]
vége

eljárás ábra :h
  tollatle ismétlés 4 [rombusz :h/6 jobbra 90] tollatfel
vége

eljárás rombusz :h
  balra 30 ismétlés 2 [előre :h jobbra 60 előre :h jobbra 120]
  jobbra 20 tollatfel előre :h tölt hátra :h
  jobbra 20 előre :h tölt hátra :h balra 10 tollatle
vége

```

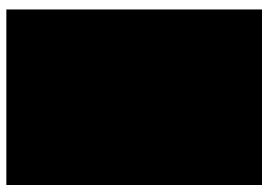
## Eltűnő és előtűnő elemek

A fiziológiai illúziók, mint a ragyogó fényeket követő utóképek vagy a túlzottan hosszan váltakozó minták adaptálódó ingerei (feltételes perceptuális utóhatás), feltehetőleg hatást gyakorolnak a szemek és az agy egy jellegzetes típusának túlzott ingerlésére – világosság, dőlés, szín, mozgás stb.

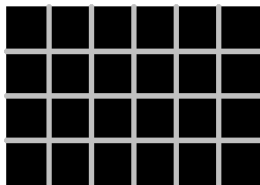
### Vibráló pontok

Készítsd el az alábbi, optikai csalódást mutató ábrákat rajzoló Logo eljárásokat (tégla :a :b, rács :dba :dbb :t, pöttyök :dba :dbb :t)! A tégla :a :b eljárás :a\*:b méretű fekete téglalapot rajzoljon! A rács :dba :dbb :t eljárás :dba\*:dbb méretű szürke rácsot rajzoljon 5 egység vastagságú vonalakkból, ahol a rácsvonalak :t távolságra vannak egymástól! A pöttyök :dba :dbb :t eljárás :dba\*:dbb méretű fehér pöttymintát rajzoljon 10 egység nagyságú pontokból, ahol a pöttyök :t távolságra vannak egymástól!

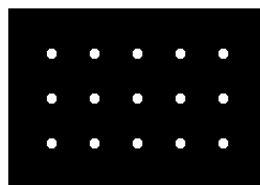
Az alábbi képeket egy vagy több eljárás egymás utáni hívásával kaptuk:



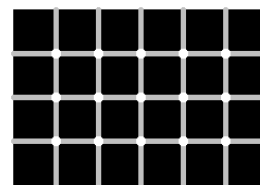
tégla 60 90



tégla 60 90  
rács 3 5 15



tégla 60 90  
pöttyök 3 5 15



tégla 60 90  
rács 3 5 15  
pöttyök 3 5 15

A feladat megoldása egy fekete téglalap, továbbá két speciális (szürke, illetve fehér) mozaik kirajzolása.

```
eljárás pöttyök :n :m :t
  tollatfel tollvastagság! 10 tollszín! 15
  ismétlés :n [előre :t jobbra 90
              ismétlés :m [előre :t tollatle előre 0 tollatfel]
              hátra :m*:t balra 90]
  hátra :n*:t tollatle
vége
```

```
eljárás rács :n :m :t
  tollvastagság! 5 tollszín! 7
  ismétlés :m [tollatfel jobbra 90 előre :t balra 90 tollatle
              előre (:n+1)*:t hátra (:n+1)*:t]
  tollatfel jobbra 90 hátra :t*:m balra 90 tollatle
  ismétlés :n [tollatfel előre :t tollatle jobbra 90
              előre (:m+1)*:t hátra (:m+1)*:t balra 90]
  tollatfel hátra :t*:n tollatle
vége
```

```
eljárás tégla :x :y
  tollvastagság! 1 tollszín! 0
  ismétlés 2 [előre :x jobbra 90 előre :y jobbra 90]
  tollatfel jobbra 45 előre 3 tölt hátra 3 balra 45 tollatle
vége
```

### Mozgás látszata

Az alábbi mozaik sárga alapon elhelyezett mandulaszerű elemekből áll (a mandulákat sokan levélnek látják, de ez a lényegen nem változtat). *Ha valaki az ábrán mozgást lát, az csak érzékcsalódás!*

## Forgó mandulák

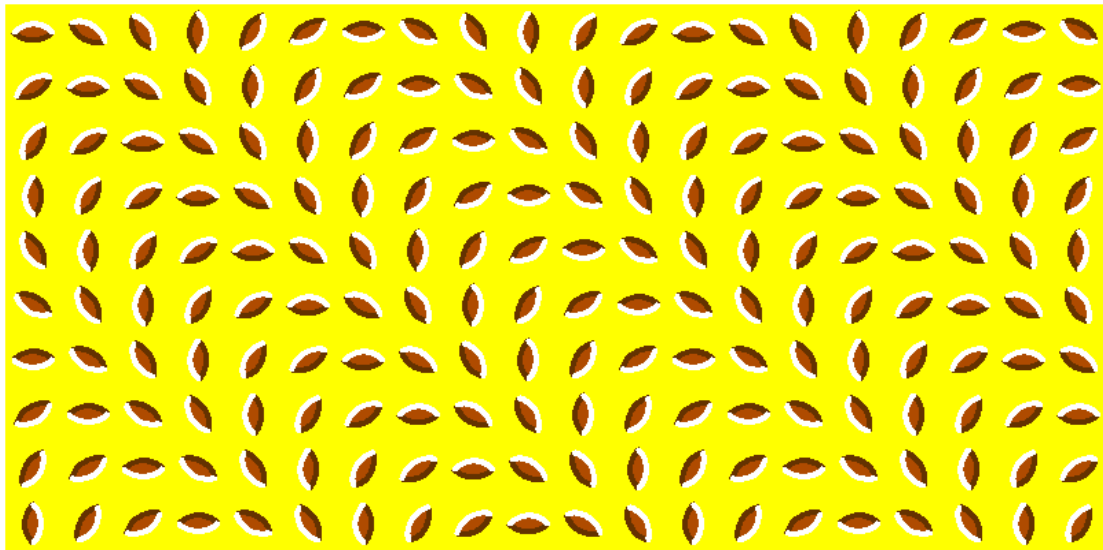
A mandula :h :szög eljárás egy :h oldalhosszúságú négyzetbe rajzoljon egy mandulát, amelynek középvonala az y-tengellyel :szög fokos szöget zárjon be! A mandula világosbarna, egyik oldalán vastag barna, a másik oldalán pedig vastag fehér széle legyen! A sor :m :h :szög eljárás :m mandulát rajzoljon egymás mellé, a legelső :szög fokos szögben álljon! A mozaik :n :m :h eljárás :n sort rajzoljon egymás fölé!



mandula 40 0



sor 8 40 0



mozaik 10 20 10

```

eljárás mozaik :n :m :h
  tollszín! 14 tolllatle
  ismétlés 2 [előre :n*:h jobbra 90 előre :m*:h jobbra 90]
  tollatfel jobbra 45 előre :h töltőszín! 14 tölt hátra :h
  balra 45 előre :h/2 jobbra 90 előre :h/2 balra 90 tollatle
  sorok :n-1 :m :h 0
  tollatfel jobbra 90 hátra :h/2 balra 90 hátra :h/2 tollatle
vége

eljárás sor :m :h :fok
  jobbra :fok mandula :h/2 balra :fok
  ha :m>1 [tollatfel jobbra 90 előre :h balra 90 tollatle
    sor :m-1 :h :fok+30
    tollatfel jobbra 90 hátra :h balra 90 tollatle]
vége

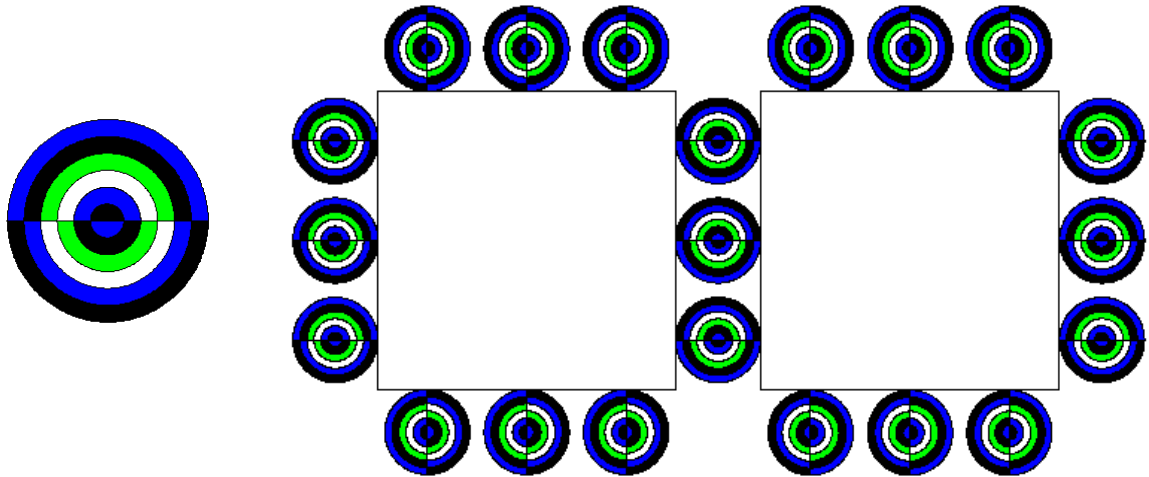
eljárás sorok :n :m :h :fok
  sor :m :h :fok
  ha :n>0 [tollatfel előre :h tollatle sorok :n-1 :m :h :fok+30]
vége

eljárás mandula :r
  tollatfel hátra :r/2*gyök 2 tollatle tollvastagság! 4
  tollszín! [100 50 0] balra 45 ív 90 :r jobbra 90
  tollszín! 15 ív 90 :r jobbra 135
  tollvastagság! 1 tollatfel előre :r töltőszín! [175 75 0]
  tölt hátra :r előre :r/2*gyök 2 tollatle
vége
    
```

```
eljárás ív :fok :r
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége
```

### Forgó körök

Ha sokáig nézed a jobb oldali mellékelt ábrát, akkor úgy látszik, mintha forognának a körök. Készíts köröm :r és forgó :hossz eljárást, ahol a :r a legnagyobb kör sugara, a :hossz pedig a négyzetek oldala!



köröm 100

forgó 200

```
eljárás köröm :r
  kör2 :r/6 "fekete "kék kör2 :r/3 "kék "fekete
  kör2 :r/2 "zöld "fehér kör2 :r/3*2 "fehér "zöld
  kör2 :r/6*5 "fekete "kék kör2 :r "kék "fekete
vége
```

```
eljárás kör2 :r :a :b
  jobbra 90 előre :r balra 90
  ismétlés 360 [előre :r*3,14159/180 jobbra 1]
  jobbra 90 hátra :r balra 90
  tollatfel előre :r/2 tollszín! :a tölt hátra :r
  tollszín! :b tölt előre :r/2 tollatle tollszín! "fekete
vége
```

```
eljárás forgó :oldal
  négyzet :oldal tollatfel jobbra 90 előre :oldal*9/7 balra 90
  tollatle előre :oldal jobbra 90 négyzet2 :oldal
  tollatfel jobbra 90 hátra :oldal*9/7 balra 90
vége
```

A négyzet és négyzet2 eljárásokban csak az oldalak számában van különbség! Összevonhatunk volna egyetlen eljárásban is.

```
eljárás négyzet :oldal
  ismétlés 4 [ismétlés 3 [előre :oldal/6 balra 90 előre :oldal/7 jobbra 90
    köröm :oldal/7 balra 90 hátra :oldal/7 jobbra 90
    előre :oldal/6 jobbra 90]
vége
```

```
eljárás négyzet2 :oldal
  ismétlés 3 [ismétlés 3 [előre :oldal/6 balra 90 előre :oldal/7 jobbra 90
    köröm :oldal/7 balra 90 hátra :oldal/7 jobbra 90
    előre :oldal/6 jobbra 90]
vége
```

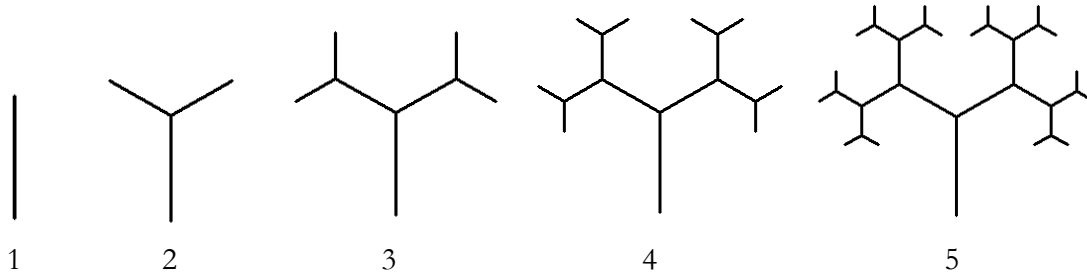
## Variációk fa rajzolásra

Alapfeladat: Egy fa egy  $:H$  hosszúságú törzsből áll, amelynek végén szimmetrikusan, egymással 120 fokos szöget bezárva egy-egy újabb fa nő ki. A fa gyökerétől az ágak végéig  $:N$  lépést lehet megtenni, a törzsből kinövő újabb fák törzhossza az eredeti fa törzhosszának kétharmada.

```

eljárás fa :n :h
  előre :h
  ha :n>1 [balra 60 fa :n-1 :h*2/3 jobbra 120 fa :n-1 :h*2/3 balra 60]
  hátra :h
vége

```



Feladatsor az alapmegoldás programjából kiindulva:

- csak a paraméterek változnak (szögek, hosszak)
- hívások száma változik (ágszám nem 2)
- szinttől függő új tulajdonságok: vastagság, szín (utolsó ág más, folyamatos változás), ág-szám

Feladatsor az alapmegoldás eredményéből kiindulva:

- nem növekvő, de sűrűsödő fák
- a törzs (vagy annak része) is fa, az ág egy része nem fa
- alapelem két oldalon rajzolva (pl. kaktusz, speciális törzsű fák)

Feladatsor két megoldás kombinálásával:

- kétféle fa hívja egymást (más ágszámú, más szögű, más ághosszú), bal- és jobboldali fa más
- az ágak bal- és jobboldalát külön rajzoljuk meg

Feladatsor külső hatásokkal befolyásolva:

- méret és irányfüggő változások, irányfüggő görbe vonalakkal, helyfüggő (pl. ne érjen a gyökere alá)
- a fa nem szimmetrikus (struktúrát leíró listával paraméterezett)
- véletlen fák (véletlen növekedés, véletlen ághossz, véletlen ...)

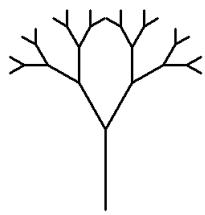
Feladatsor más alapelemekből:

- törzhossz helyett alapelem szám

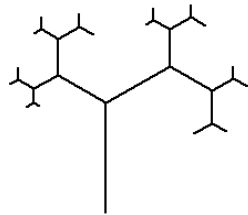
### 1. Feladatsor az alapmegoldás programjából kiindulva

A módszer lényege, hogy az alapfeladat megoldását sok kis lépésben módosítsuk úgy, hogy újabb feladatokat állítsunk elő belőle. Az előállított feladatok lehetnek az alapfeladathoz hasonló nehézségűek, de lehetnek más nehézségi csoportba tartozók is.

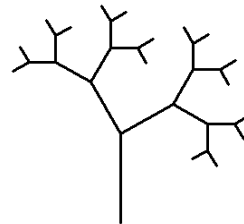
**Elemi feladatvariációk:**



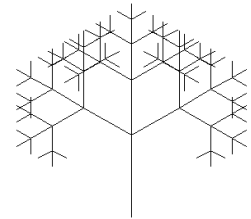
1. variáció



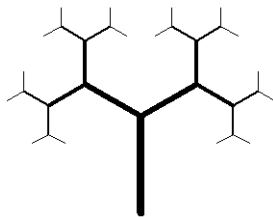
2. variáció



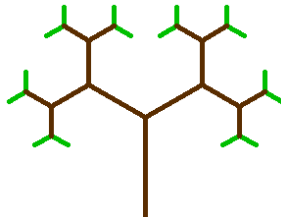
3. variáció



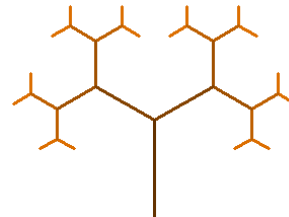
4. variáció



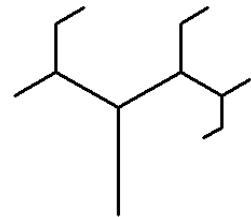
5. variáció



6. variáció



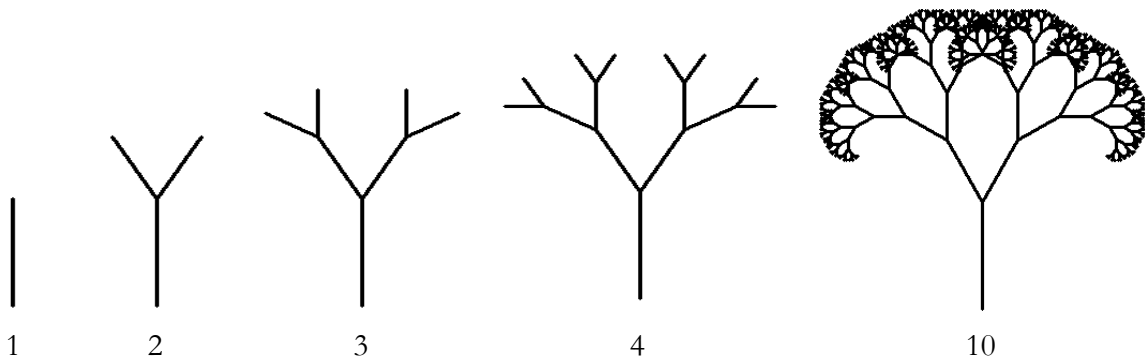
7. variáció



8. variáció

Nézzük meg, hogy a különböző képeknek milyen programváltozat felel meg (az alapváltozathoz képesti változások vastagon szedettek)!

1. variáció: Mások legyenek az ágak közötti szögek!

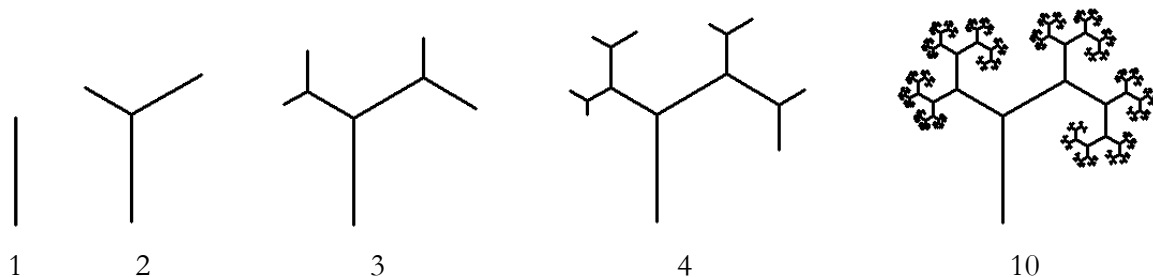


A megoldás a három elfordulás megváltoztatása.

```

eljárás fa1 :n :h
  előre :h
  ha :n>1 [balra 30 fa1 :n-1 :h*2/3 jobbra 60 fa1 :n-1 :h*2/3 balra 30]
  hátra :h
vége
    
```

2. variáció: Balra, illetve jobbra más arányúak legyenek a törzshosszak!

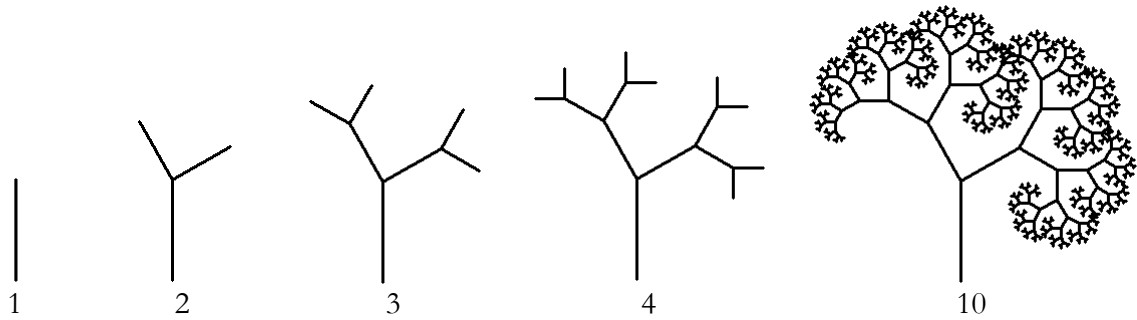


A megoldásban a ét eljáráshívás paramétereit változtatjuk meg.

```

eljárás fa2 :n :h
  előre :h
  ha :n>1 [balra 60 fa2 :n-1 :h/2 jobbra 120 fa2 :n-1 :h*2/3 balra 60]
  hátra :h
vége
  
```

3. variáció: Az ágak szögeit változtassuk meg!

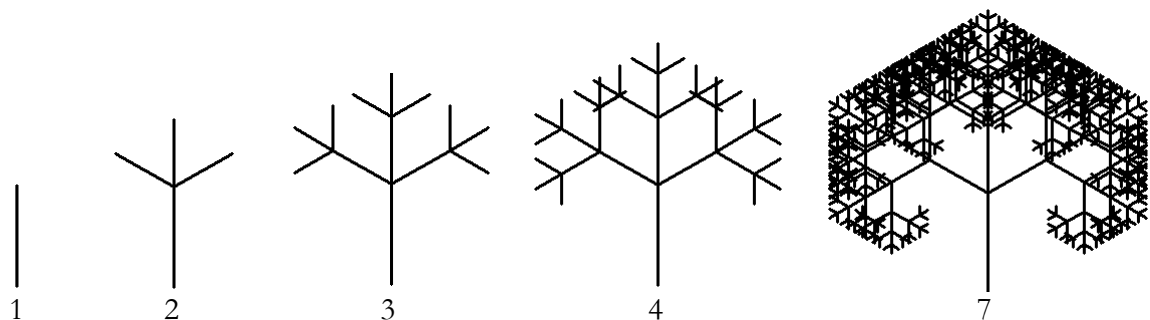


A megoldás a három elfordulás megváltoztatása.

```

eljárás fa3 :n :h
  előre :h
  ha :n>1 [balra 30 fa3 :n-1 :h*2/3 jobbra 90 fa3 :n-1 :h*2/3 balra 60]
  hátra :h
vége
  
```

4. variáció: A az ágak számát változtassuk meg!

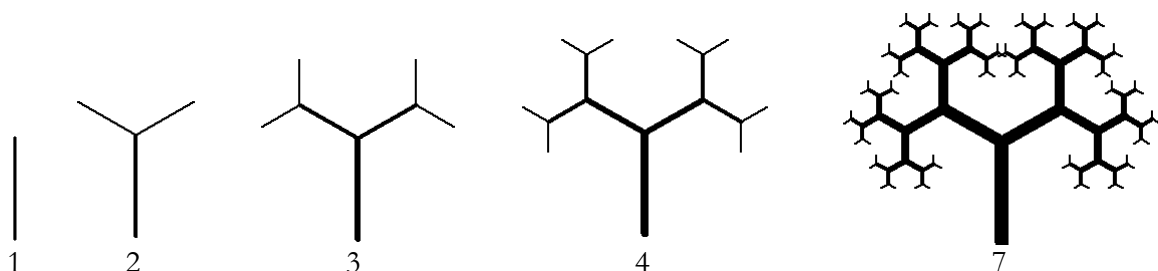


A megoldásban háromszor hívjuk rekurzívan a rajzolást.

```

eljárás fa4 :n :h
  előre :h
  ha :n>1 [balra 60 fa4 :n-1 :h*2/3 jobbra 60
          fa4 :n-1 :h*2/3 jobbra 60 fa4 :n-1 :h*2/3 balra 60]
  hátra :h
vége
  
```

5. variáció: Az ágak vastagságát változtassuk meg!



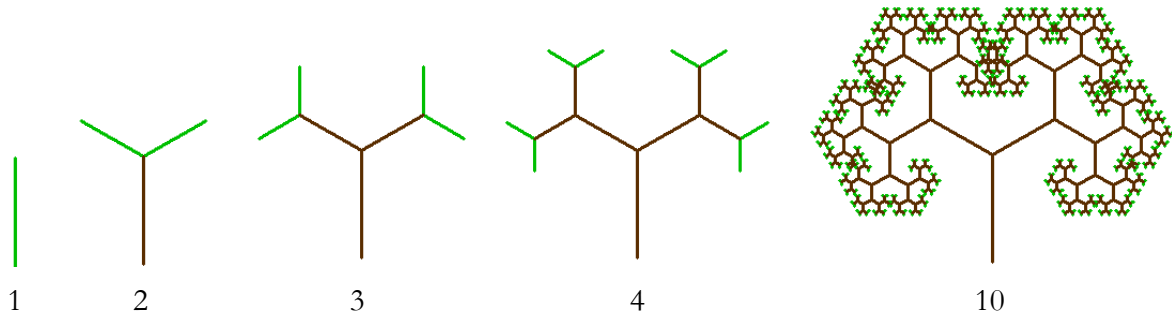


A törzsön előre haladva (és hátra is) változtassuk meg a tollvastagságot!

```

eljárás fa5 :n :h
  tollvastagság! :n*2 előre :h
  ha :n>1 [balra 60 fa5 :n-1 :h*2/3 jobbra 120 fa5 :n-1 :h*2/3 balra 60]
  tollvastagság! :n*2 hátra :h
vége
    
```

6. variáció: Az ágak színét változtassuk meg; az utolsó zöld, a többi barna színű legyen!

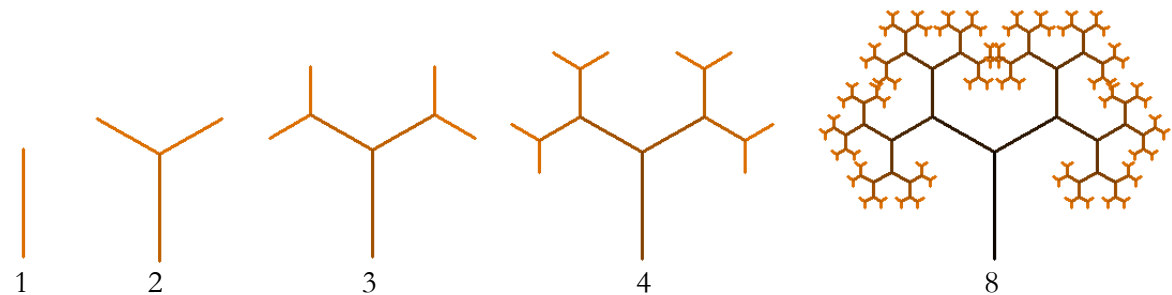


Ahogy az előző feladatban a tollvastagságot, most a tollszínt kell megváltoztatni! A tollszín megadható színpaletta kóddal és RGB-kóddal is.

```

eljárás fa6 :n :h
  hakülönben :n=1 [tollszín! 2] [tollszín! [90 45 0]] előre :h
  ha :n>1 [balra 60 fa6 :n-1 :h*2/3 jobbra 120 fa6 :n-1 :h*2/3 balra 60]
  hakülönben :n=1 [tollszín! 2] [tollszín! [90 45 0]] hátra :h
vége
    
```

7. variáció: Az ágak színét változtassuk meg, az ágak felé haladva egyre világosabb legyen!

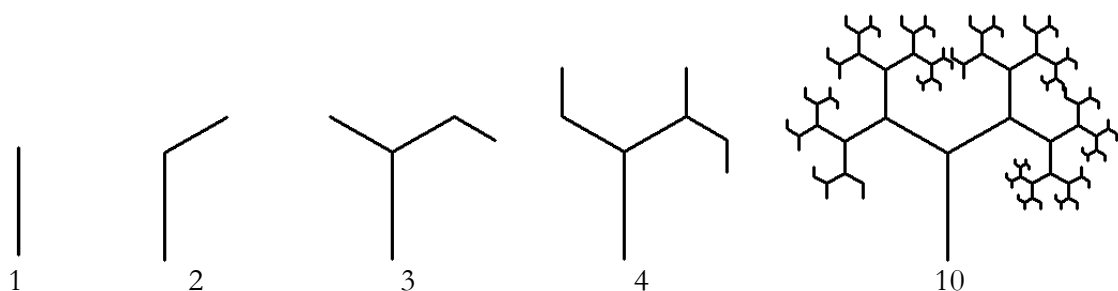


Itt a rajzolószínnek a szintek számától kell függeni! A megoldás képlete csak :n=8-ig jó.

```

eljárás fa7 :n :h
  tollszín! (lista 250-30*:n 125-15*:n 0) előre :h
  ha :n>1 [balra 60 fa7 :n-1 :h*2/3 jobbra 120 fa7 :n-1 :h*2/3 balra 60]
  tollszín! (lista 250-30*:n 125-15*:n 0) hátra :h
vége
    
```

8. variáció: A törzstől a levélhez jutás lépésszámát változtassuk meg!



A jobboldali ág 1, a baloldali ág 2 szinttel kisebb, mint a teljes fa szintszáma. Vigyázni kell arra, hogy 0 szintű fát ne rajzoljunk!

```

eljárás fa8 :n :h
  ha :n>0 [előre :h]
  ha :n>1 [balra 60 fa8 :n-2 :h*2/3 jobbra 120 fa8 :n-1 :h*2/3 balra 60]
  ha :n>0 [hátra :h]
vége
    
```

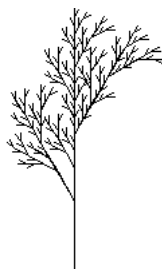
Ez a 8 variáció természetesen bármilyen módon kombinálva is alkalmazható, azaz lehet például olyan fát rajzolni, amely háromfelé ágazik, a balra növe ágak rövidebbek, s a fán felfelé haladva az ágak vastagsága csökken.

## 2. Feladatsor az alapgondolat eredményéből kiindulva:

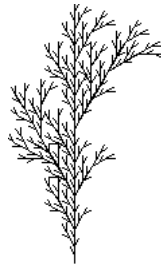
A megoldás programja helyett kiindulhatunk a megoldás eredményéből, azaz most a képből is. Ez viszonylag ritkaság a programozás világában, akkor van rá lehetőség, ha az eredmény elég összetett (sokszor grafikus eredménynél lehet, bár ez nem feltétlenül szükséges). Mivel az eredmény itt maga a feladat, ezért ez a módszer abba a csoportba tartozik, amikor a feladatot módosítjuk.

### Ágak a törzsből

Első próbálkozásként a törzsből is nőjenek ki valahol ágak!



9. variáció

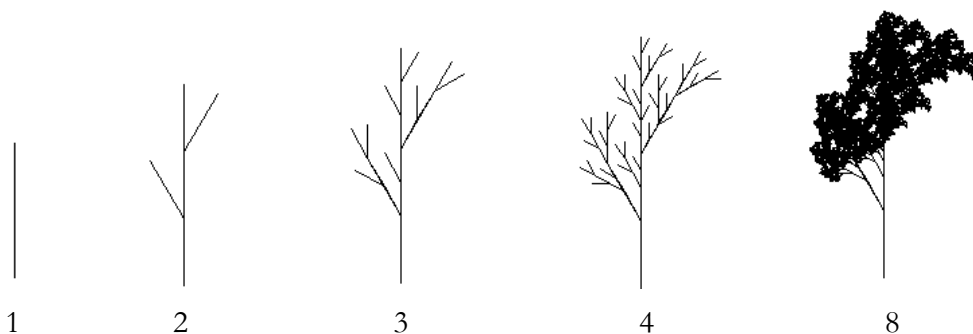


10. variáció



11. variáció

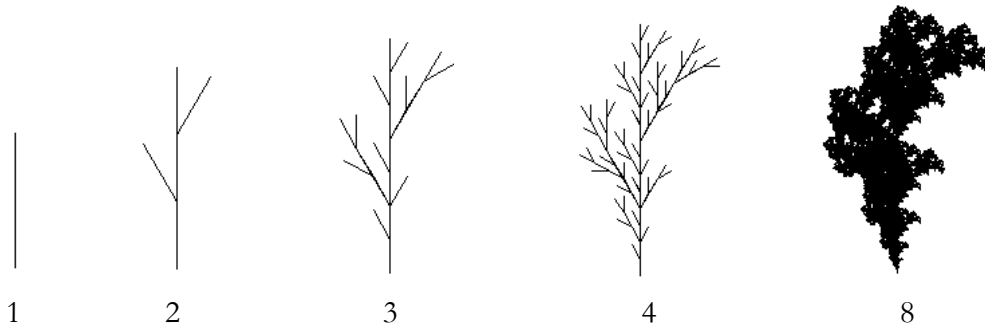
9. variáció: A törzs egyik fele legyen állandó, a másik fele pedig ág! Az ága aljából balra, az ág végéből pedig felfelé, illetve jobbra is nő újabb ág.



```

eljárás fa9 :n :h
  hakülönben :n=1 [előre :h hátra :h]
  [előre :h/2 balra 30 fa9 :n-1 :h/2 jobbra 30 fa9 :n-1 :h/2
  előre :h/2 jobbra 30 fa9 :n-1 :h/2 balra 30 fa9 :n-1 :h/2 hátra :h]
vége
    
```

10. variáció: A törzs legyen két ág! Az alsó ág végéből balra, a felső ág végéből pedig felfelé, illetve jobbra is nő újabb ág.



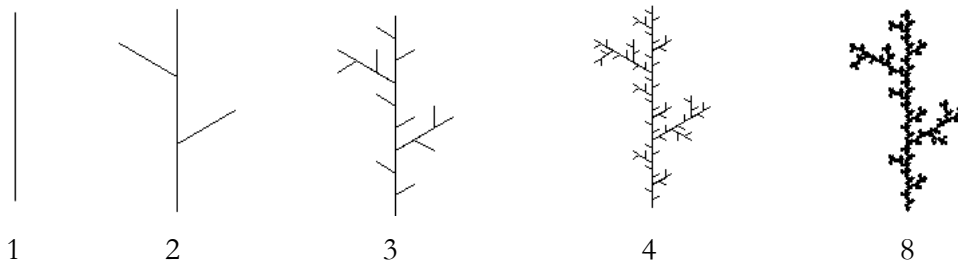
```

eljárás fa10 :n :h
  hakülönben :n=1 [előre :h hátra :h]
  [fa10 :n-1 :h/2 előre :h/2 balra 30 fa10 :n-1 :h/2 jobbra 30
   fa10 :n-1 :h/2 előre :h/2 jobbra 30 fa10 :n-1 :h/2 balra 30
   fa10 :n-1 :h/2 hátra :h]

```

vége

11. variáció: A fa 5 ágból álljon, melyek közül három a törzsből keletkezik, kettő pedig ebből ágazik el!



```

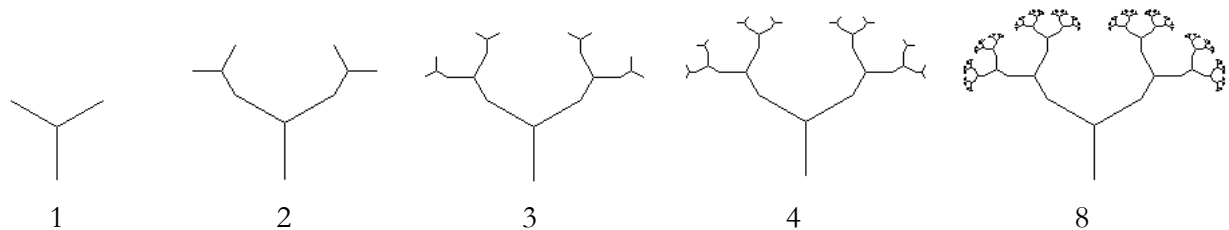
eljárás fall :n :h
  hakülönben :n=1 [előre :h hátra :h]
  [fall :n-1 :h/3 előre :h/3 jobbra 60 fall :n-1 :h/3 balra 60
   fall :n-1 :h/3 előre :h/3 balra 60 fall :n-1 :h/3 jobbra 60
   fall :n-1 :h/3 hátra 2*:h/3]

```

vége

**Az ágak egy része nem fa**

12. variáció: Olyan fa, ami nem törzsből és fákból, hanem kettéágazó törzsből és a végükön kinövő fákból áll:



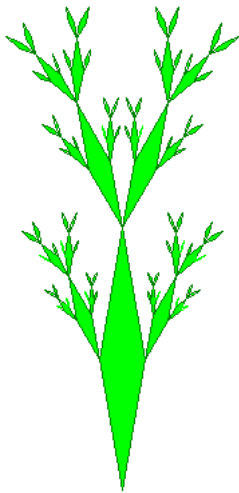
```

eljárás fa12 :n :h
  előre :h balra 60 előre :h
  ha :n>1 [jobbra 30 fa12 :n-1 :h/2 balra 30]
  hátra :h jobbra 120 előre :h
  ha :n>1 [balra 30 fa12 :n-1 :h/2 jobbra 30]
  hátra :h balra 60 hátra :h

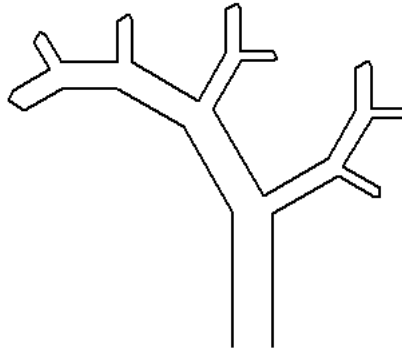
```

vége

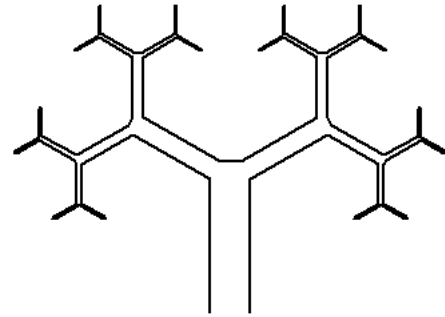
Ágak oldalai külön rajzolva



13. variáció

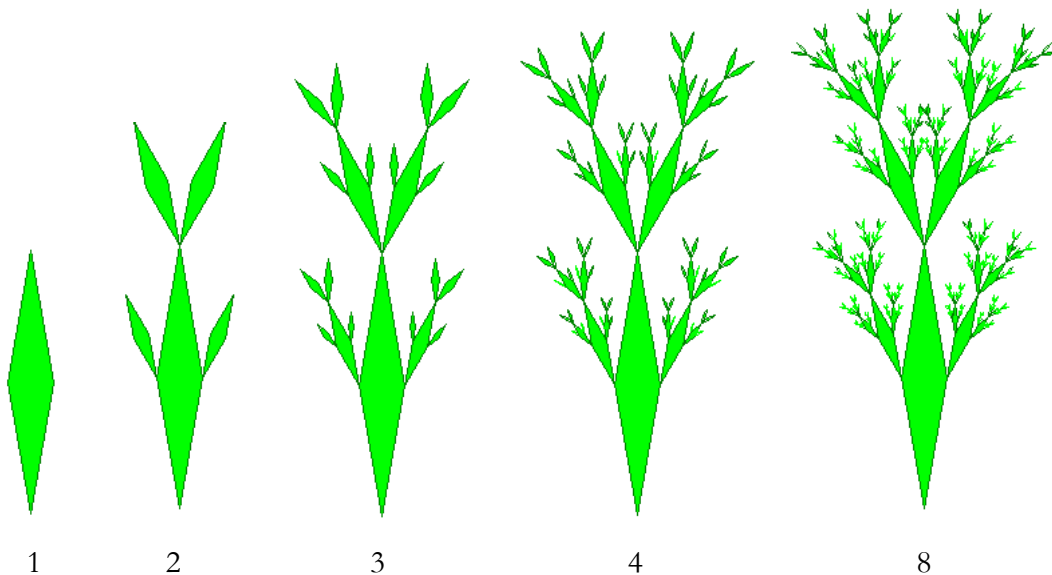


14. variáció



15. variáció

13. variáció: A fa törzse rombusz. A törzs két oldalsó sarkából egy-egy, a tetejéből pedig 2 ág nőhet ki.

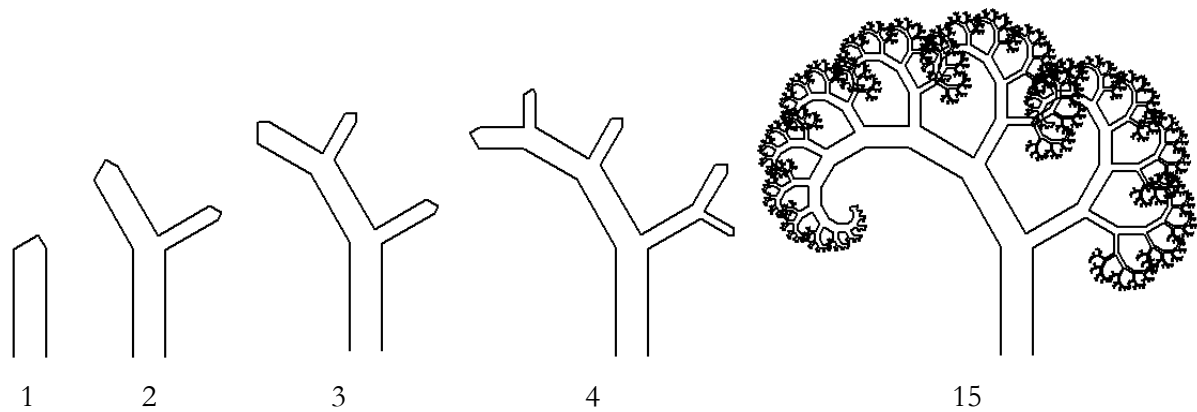


A kaktusznál – a belseje festésén túl – az az újdonság, hogy az oldalából is nőnek ki újabb ágak.

```

eljárás fa13 :n :h
  tollszín! [0 128 0]
  balra 10 előre :h balra 10 ha :n>1 [fa13 :n-1 :h/3]
  jobbra 30 előre :h balra 30 ha :n>1 [fa13 :n-1 :h/2]
  jobbra 40 ha :n>1 [fa13 :n-1 :h/2]
  jobbra 150 előre :h balra 150 ha :n>1 [fa13 :n-1 :h/3]
  jobbra 170 előre :h jobbra 170
  töltőszín! [0 255 0] tollatfel előre :h tölt hátra :h tollatle
vége
    
```

14. variáció: A fa törzse :v szélességű téglalap. A baloldali ág ehhez 30, a jobboldali ág pedig 60 fokos szögben csatlakozik.

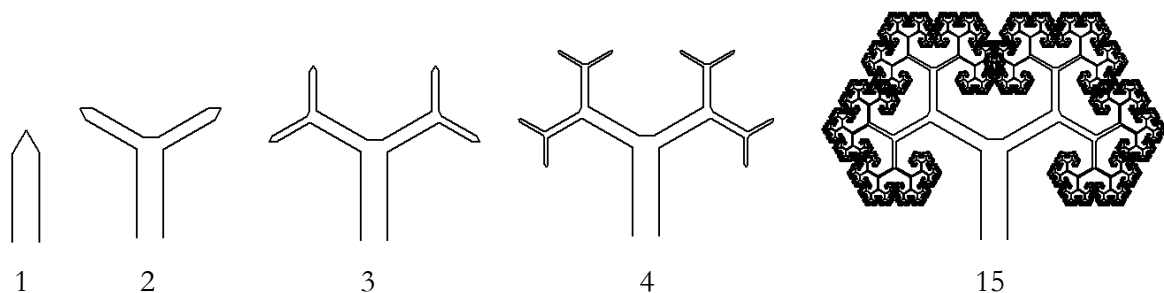


A két szélesebb törzsű fa jellemzője – a kaktuszéval együtt – az, hogy nem ugyanazon a vonalon jövünk vissza az ágak megrajzolása után, mint amin az ágak rajzolása előtt haladtunk. Ezeknél a fáknál emiatt új paraméterként jelenik meg a fa vastagsága is. A baloldali ág 1. a jobboldali pedig 2 szinttel kisebb a fa szintjénél, de most a 0. és az 1. szintű fa is egy ágból áll.

```

eljárás fa14 :n :h :v
  előre :h
  ha :n<=1 [jobbra 60 előre :v*cos 30 jobbra 90 előre :v*sin 30 jobbra 30]
  ha :n>1 [balra 30 fa14 :n-1 :h*3/4 :v*cos 30 balra 90
           fa14 :n-2 :h*3/4*3/4 :v*sin 30 balra 60]
  előre :h
vége
    
```

**15. variáció:** A fa törzse :v szélességű téglalap. A törzshöz mindkét ág 60 fokos szögben csatlakozik.

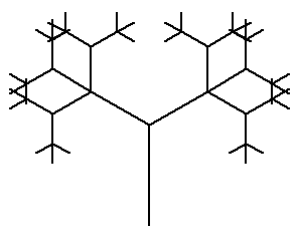


```

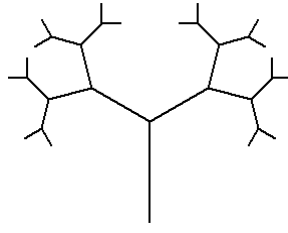
eljárás fa15 :n :h :v
  előre :h
  ha :n<=1 [jobbra 30 előre :v jobbra 120 előre :v jobbra 30]
  ha :n>1 [balra 60 fa15 :n-1 :h*2/3 :v/2 balra 30 előre :v/2 balra 30
           fa15 :n-1 :h*2/3 :v/2 balra 60]
  előre :h
vége
    
```

### 3. Feladatsor két megoldás kombinálásával:

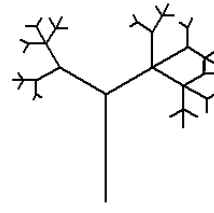
Rekurziót alkalmazó megoldásoknál lehetőségünk van a közvetett rekurzióra. Sok esetben olyan fát kell rajzolni, amelyik szintenként valamilyen más jellemzővel rendelkezik. Ilyenkor kettő vagy több farajzoló eljárást írunk, amelyek egymást hívják.



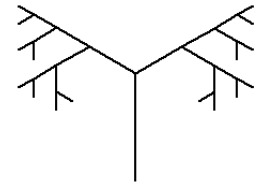
16. variáció



17. variáció

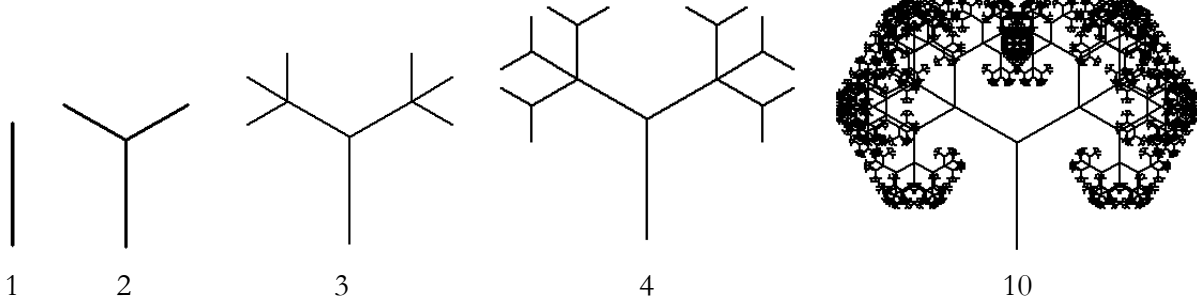


18. variáció



19. variáció

16. variáció: a fán felváltva 2-, illetve 3-ágú elágazások legyenek!



Két eljárást írunk. A kétágú fa háromágú fát hív, a háromágú fa pedig kétágút.

eljárás fa16a :n :h

előre :h

ha :n>1 [balra 60 fa16b :n-1 :h\*2/3 jobbra 120  
fa16b :n-1 :h\*2/3 balra 60]

hátra :h

vége

eljárás fa16b :n :h

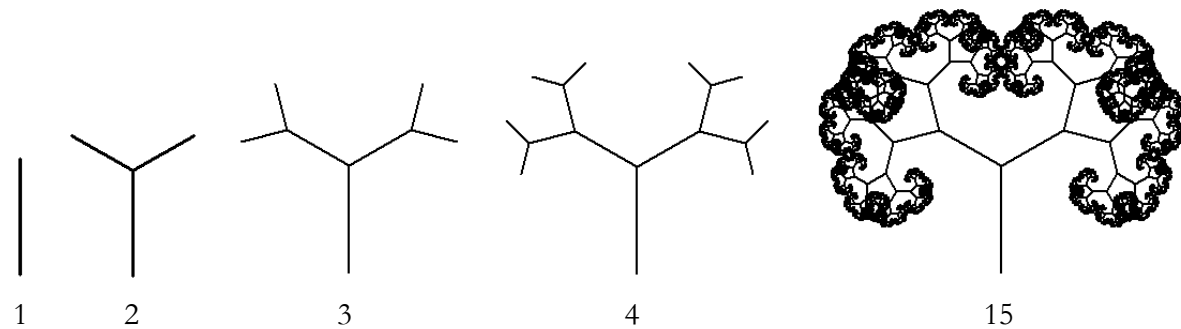
előre :h

ha :n>1 [balra 60 fa16a :n-1 :h\*2/3 jobbra 60  
fa16a :n-1 :h\*2/3 jobbra 60 fa16a :n-1 :h\*2/3 balra 60]

hátra :h

vége

17. variáció: a fán felváltva 120, illetve 90 fokos szögben találkozzanak az ágak!



Két eljárást írunk. A 120 fokos szögű fa 90 fokos fát hív, a 90 fokos fa pedig 120 fokosat.

eljárás fa17a :n :h

előre :h

ha :n>1 [balra 60 fa17b :n-1 :h\*2/3 jobbra 120  
fa17b :n-1 :h\*2/3 balra 60]

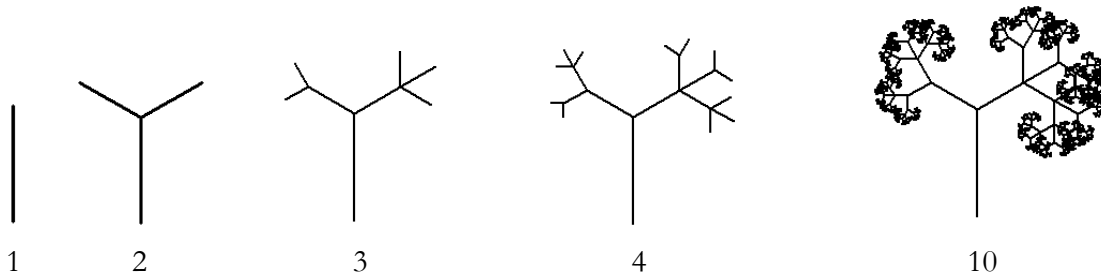
hátra :h

vége

```

eljárás fa17b :n :h
  előre :h
  ha :n>1 [balra 45 fa17a :n-1 :h*2/3 jobbra 90
           fa17a :n-1 :h*2/3 balra 45]
  hátra :h
vége
    
```

18. variáció: a középső, a bal és a jobboldali fa is más!



Itt már három eljárást kell írni! A középső fa bal- és jobboldali fát hív. A baloldali fa bal- és jobboldali fát hív. A jobboldali fa pedig baloldali, középső és jobboldali fát is.

```

eljárás fa18 :n :h
  előre :h
  ha :n>1 [balra 60 fa18b :n-1 :h/2 jobbra 120 fa18j :n-1 :h/2 balra 60]
  hátra :h
vége
    
```

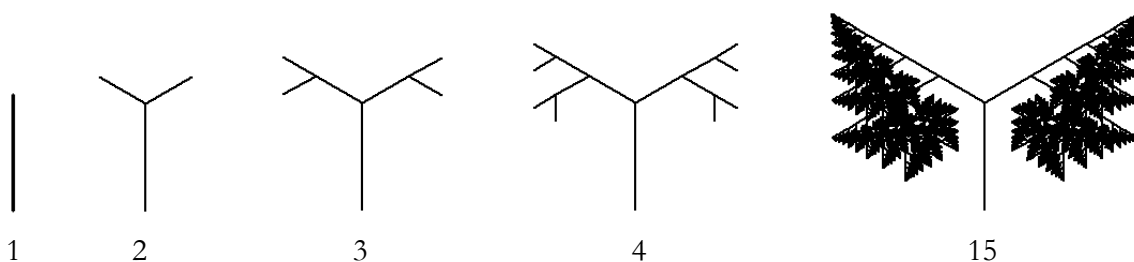
```

eljárás fa18b :n :h
  előre :h
  ha :n>1 [balra 60 fa18b :n-1 :h/2 jobbra 90 fa18j :n-1 :h/2 balra 30]
  hátra :h
vége
    
```

```

eljárás fa18j :n :h
  előre :h
  ha :n>1 [balra 60 fa18b :n-1 :h*2/3 jobbra 60
           fa18 :n-1 :h*3/4 jobbra 60 fa18j :n-1 :h*2/3 balra 60]
  hátra :h
vége
    
```

19. variáció: a fa baloldali, és jobboldali fából áll!



Itt is három eljárást kell írni, de csak a baloldali és a jobboldali fa rekurzív és mindegyik csak saját magát hívja!

```

eljárás fa19 :n :h
  előre :h
  ha :n>1 [balra 60 fa19b :n-1 :h/2 jobbra 120 fa19j :n-1 :h/2 balra 60]
  hátra :h
vége
    
```

eljárás fa19b :n :h

előre :h

ha :n>1 [balra 60 fa19b :n-1 :h\*0,7 jobbra 60 fa19b :n-1 :h\*0,7]

hátra :h

vége

eljárás fa19j :n :h

előre :h

ha :n>1 [jobbra 60 fa19j :n-1 :h\*0,7 balra 60 fa19j :n-1 :h\*0,7]

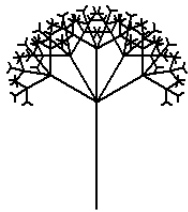
hátra :h

vége

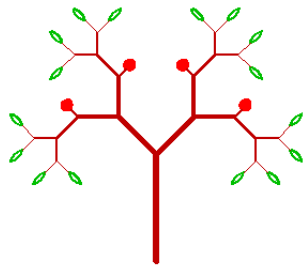
#### 4. Feladatsor külső hatásokkal befolyásolva:

##### Paraméterfüggő rajzok

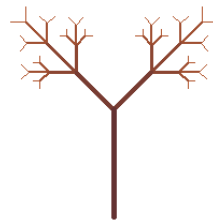
A következő variációkban a hatás a paraméterek értékétől (évtől vagy hosszól) függ.



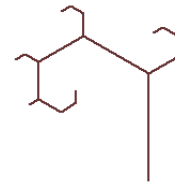
20. variáció



21. variáció



22. variáció

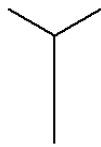


23. variáció

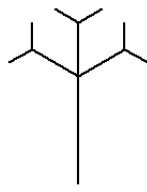
20. variáció: legyen szintenként különböző számú ág!



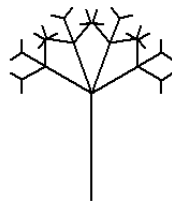
1



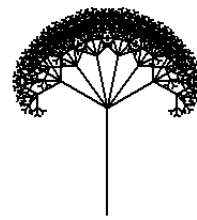
2



3



4



6

eljárás fa20 :n :h

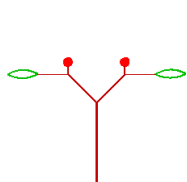
előre :h

ha :n>1 [balra 60 ismétlés :n-1 [fa20 :n-1 :h/2 jobbra 120/(:n-1)  
fa20 :n-1 :h/2 balra 60]

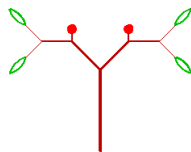
hátra :h

vége

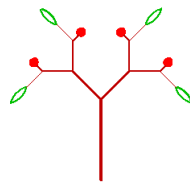
21. variáció: A gyümölcsfákat adott évben úgy metszik, hogy a fa belsejéből kiritkítják az ágakat, hogy katlan alakú legyen a közepe és besüssön oda a nap. A fa mindkét oldaláról a közepe felé eső ágakat metszik ki.



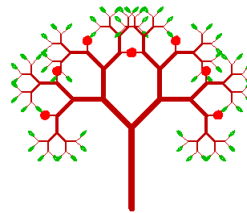
3 2



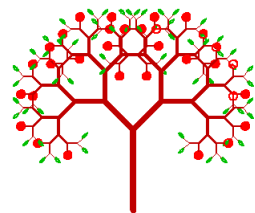
4 2



4 3



8 4



8 6



Tudnunk kell, hogy a fa baloldali vagy jobboldali ágán vagyunk-e! A két ágot megírhatnánk két külön eljárásban, de most azt a megoldást választjuk, hogy ugyanannak az egyetlen eljárásnak paramétere lesz, hogy a baloldali ágán vagyunk-e.

```

eljárás fa21 :n :mikor :h
  tollvastagság! :n tollszín! 4
  előre :h
  ha :év>1 [balra 45 fa21r :n-1 :mikor-1 :h/2 "igaz
            jobbra 90 fa21r :n-1 :mikor-1 :h/2 "hamis balra 45]
  tollvastagság! :n hátra :h
vége

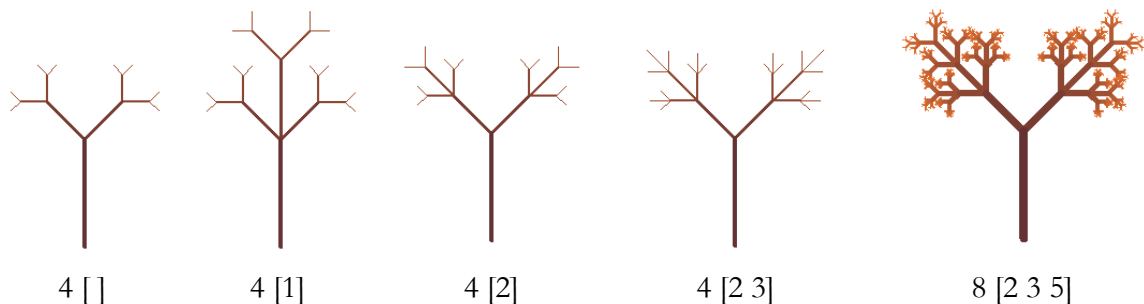
eljárás fa21r :n :mikor :h :irány
  tollvastagság! :n előre :h
  hakülönben :n>1 [balra 45
                   hakülönben és (:mikor=2) nem :irány [gyümölcs 5]
                   [fa21r :n-1 :mikor-1 3*:h/4 :irány]
                   jobbra 90
                   hakülönben és (:mikor=2) :irány [gyümölcs 5]
                   [fa21r :n-1 :mikor-1 3*:h/4 :irány] balra 45]
  [levél :h]
  tollvastagság! :n hátra :h
vége

eljárás gyümölcs :h
  tollvastagság! 1 előre 2*:h tollszín! 12 tollvastagság! 2
  balra 90 ismétlés 360 [előre :h*3,14159/180 jobbra 1]
  jobbra 90 töltőszín! 12 tollatfel előre :h tölt hátra :h tollatle
  tollszín! 4 hátra 2*:h
vége

eljárás levél :h
  tollszín! 2 tollvastagság! 2
  balra 30 ismétlés 60 [előre :h*3,14159/180 jobbra 1]
  jobbra 120 ismétlés 60 [előre :h*3,14159/180 jobbra 1]
  balra 210
  tollszín! 4
vége

```

22. variáció: Adott években a fának 3 ága nő, a többi évben pedig kettő! Az ágak hossza a törzs hosszának fele. Ha van középső ág, akkor annak a hossza a törzs hosszának háromnegyede. A fa törzsének színe a [100 50 50] barna RGB szín, vastagsága a fa kora (:n). Az ágak felé haladva a barna szín évente világosodik, azaz az RGB kód [20 10 0]-val nő. Az ágak vastagsága évente eggyel csökken.



```

eljárás fa22 :n :mikor :h
  hakülönben üres? :mikor [fa22r :n [0] :h [100 50 50]]
  [fa22r :n :mikor :h [100 50 50]]
vége

```

```

eljárás fa22r :n :mikor :h :szín
  tollvastagság! :n tollszín! :szín előre :h
  ha :n>1 [balra 45 fa22r :n-1 csökk :mikor 2*:h/4 szcs :szín jobbra 45
    ha első :mikor=1 [fa22r :n-1 csökk :mikor 3*:h/4 szcs :szín]
    jobbra 45 fa22r :n-1 csökk :mikor 2*:h/4 szcs :szín balra 45]
  tollvastagság! :n tollszín! :szín hátra :h
vége

```

```

eljárás csökk :s
  ha üres? :s [eredmény :s]
  ha üres? elsőnélküli :s [eredmény elsőnek -1+első :s []]
  ha l=első :s [eredmény csökk elsőnélküli :s]
  eredmény elsőnek -1+első :s csökk elsőnélküli :s
vége

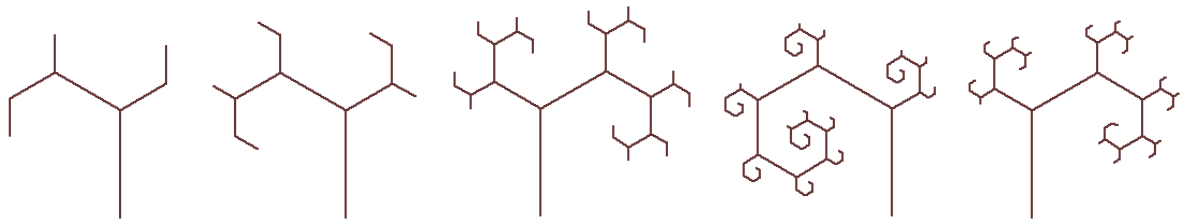
```

```

eljárás szcs :szín
  eredmény ( lista 20+első :szín 10+első elsőnélküli :szín utolsó :szín )
vége

```

23. variáció: Legyen :h a törzs hossza, minden ág végén balra :b\* :h hosszúságú, jobbra :j\* :h hosszúságú ág nő ki, de csak akkor, ha a hossza legalább :k!



100 30 0,7 0,5

100 20 0,7 0,5

100 10 0,5 0,7

100 5 0,8 0,3

100 5 0,4 0,7

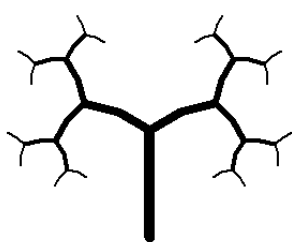
```

eljárás fa23 :h :k :b :j
  ha :h>=:k [előre :h balra 60 fa23 :h*:b :k :b :j jobbra 120
    fa23 :h*:j :k :b :j balra 60 hátra :h]
vége

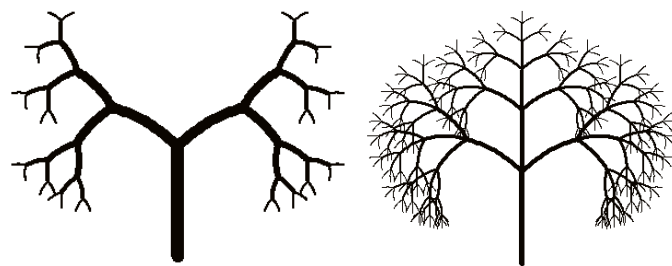
```

### Írányfüggő rajzok

Az újabb két variációban a teknőc aktuális irányától függő a fa alakulása.

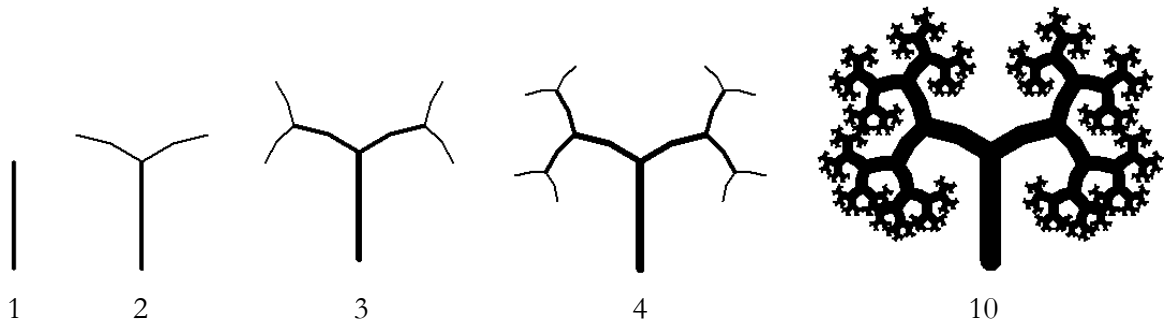


24. variáció



25. variáció 2, illetve 3 ággal

24. variáció: A fa függőlegestől balra hajló ágai közepükön 15 fokkal balra, a jobbra hajlók pedig 15 fokkal jobbra hajlanak!



```

eljárás fa24 :n :h
  tollvastagság! :n*2 előre :h/2
  hakülönben vagy egyenlő? irány 0 egyenlő? irány 180 [fa24a :n :h]
    [hakülönben irány<180 [jobbba 15 fa24a :n :h balra 15]
      [balra 15 fa24a :n :h jobbba 15]]
  tollvastagság! :n*2 hátra :h/2
vége

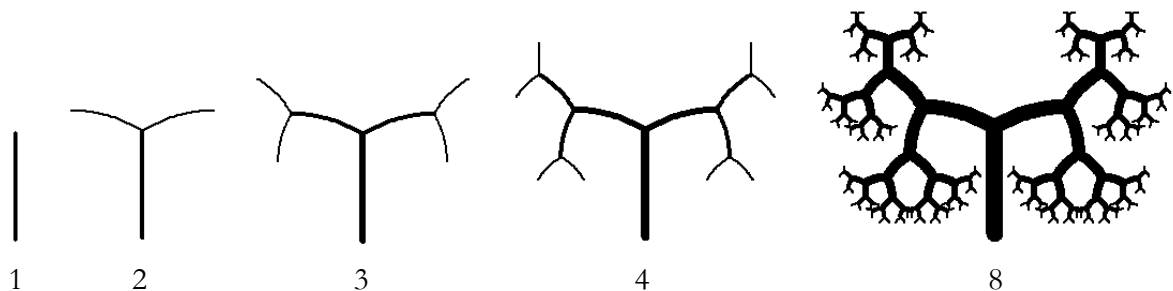
```

```

eljárás fa24a :n :h
  előre :h/2
  ha :n>1 [balra 60 fa24 :n-1 :h*2/3 jobbba 120 fa24 :n-1 :h*2/3 balra 60]
  hátra :h/2
vége

```

25. variáció: A fa függőlegestől balra hajló ágai közepükön balra, a jobbra hajlók jobbra forduló 30 fokos körívek!



```

eljárás fa25 :n :h
  tollvastagság! :n*2
  hakülönben vagy egyenlő? irány 0 egyenlő? irány 180
    [előre :h fa25a :n :h hátra :h]
    [hakülönben irány<180 [jobbív :h fa25a :n :h jobbívviSSza :h]
      [balív :h fa25a :n :h balívviSSza :h]]
  tollvastagság! :n*2
vége

```

```

eljárás fa25a :n :h
  ha :n>1 [balra 60 fa25 :n-1 :h*2/3 jobbba 120 fa25 :n-1 :h*2/3 balra 60]
vége

```

```

eljárás balív :h
  ismétlés 30 [előre :h*3,14159/90 balra 1]
vége

```

```

eljárás balívviSSza :h
  ismétlés 30 [jobbba 1 hátra :h*3,14159/90]
vége

```

```

eljárás jobbív :h
  ismétlés 30 [előre :h*3,14159/90 jobbba 1]
vége

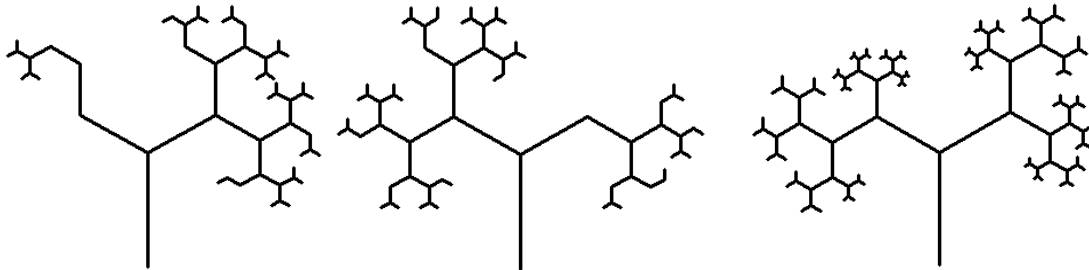
```

```

eljárás jobbívviassa :h
  ismétlés 30 [balra 1 hátra :h*3,14159/90]
vége
  
```

### Véletlenszerű fák

Az utolsó csoportban a fák véletlenszerűen növekednek.



26. variáció

27. variáció

26. variáció: Véletlentől függ, hogy balra, illetve jobbra nő-e ág.

```

eljárás fa26 :n :h
  előre :h
  ha :n>1 [balra 60 vfa26 80 :n-1 :h*2/3 jobbra 120
           vfa26 90 :n-1 :h*2/3 balra 60]
  hátra :h
vége
  
```

```

eljárás vfa26 :s :n :h
  ha véletlenszám 100<:s [fa26 :n :h]
vége
  
```

27. variáció: Véletlentől függ a balra, illetve jobbra növekvő ágak hossza.<sup>6</sup>

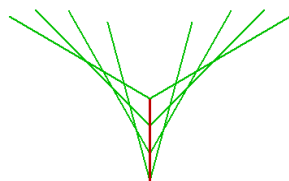
```

eljárás fa27 :n :h
  előre :h
  ha :n>1 [balra 60 fa27 :n-1 :h/2+:h/100*véletlenszám 25 jobbra 120
           fa27 :n-1 :h/2+:h/100*véletlenszám 25 balra 60]
  hátra :h
vége
  
```

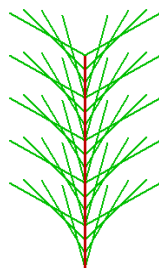
### 5. Fák más alapelemekből

A következő példákban a törzs nem szakasz, hanem valamilyen alapelemek sorozata. Emiatt nem adott hosszú előrelépésre lesz szükségünk, hanem adott darabszámú rajzolásra.

28. variáció: Az alapelemet lecseréljük egy fenyődarabra.



darab 100



fenyőág 5 50



fenyőág 20 20

<sup>6</sup> Lehetne még véletlenszerű ágyszám, véletlenszerű fordulások, véletlenszerű ágvastagság, ...

```

eljárás darab :h
  balra 15 fenyőtű :h jobbra 30 fenyőtű :h balra 15 törzs :h/3
  balra 30 fenyőtű :h jobbra 60 fenyőtű :h balra 30 törzs :h/3
  balra 45 fenyőtű :h jobbra 90 fenyőtű :h balra 45 törzs :h/3
  balra 60 fenyőtű :h jobbra 120 fenyőtű :h balra 60
vége

```

A fenyőtű és a törzs megírásával átláthatóbbá válik a darab kódja.

```

eljárás fenyőtű :h
  előre :h*2 hátra :h*2
vége

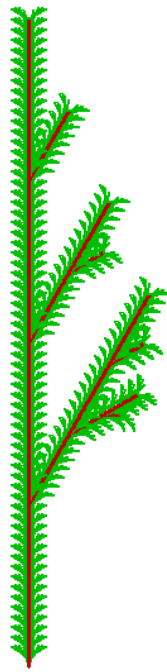
eljárás törzs :h
  tollszín! 4 tollvastagság! 3 előre :h tollvastagság! 2 tollszín! 2
vége

eljárás fenyőág :n :h
  tollszín! 2 tollvastagság! 2
  ismétlés :n [darab :h]
  tollvastagság! 1 tollszín! 0 tollatfel hátra :n*:h tollatle
vége

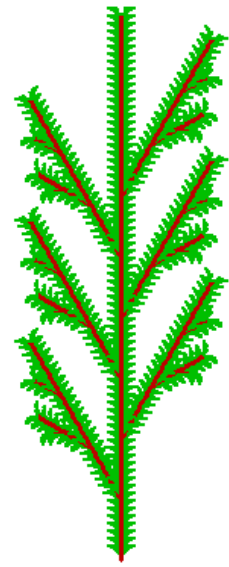
```



fenyőb 2 16 10



fenyőj 3 16 8



fenyő 3 8 5

```

eljárás fenyőb :db :n :h
  ha :db>=0 [fenyőág :n :h
    ha :db>0 [tollatfel előre :h*:n tollatle]
    balra 30 fenyőb :db-1 :n/2 :h jobbra 30
    fenyőb :db-1 :n :h
    ha :db>0 [tollatfel hátra :h*:n tollatle]]
vége

```

```

eljárás fenyőj :db :n :h
  ha :db>=0 [fenyőág :n :h
    ha :db>0 [tollatfel előre :h*:n tollatle]
    jobbra 30 fenyőj :db-1 :n/2 :h balra 30
    fenyőj :db-1 :n :h
    ha :db>0 [tollatfel hátra :h*:n tollatle]]
vége

```

vége

A fenyő két oldalán kicsit elcsúsztatva egymástól megjelennek a már megírt bal és jobb oldali fenyőrészek.

```
eljárás fenyő :db :n :h
  ismétlés :db [fenyőág :n/2 :h tollatfel előre :h*:n/2
                tollatle balra 30 fenyőb :db :n/2 :h
                jobbra 30 fenyőág :n/2 :h tollatfel
                előre :h*:n/2 tollatle jobbra 30
                fenyőj :db :n/2 :h balra 30]
  fenyőág :n :h
vége
```

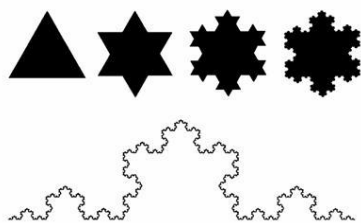
## Fraktálok

A fraktálok végtelenül komplex geometriai alakzatok, amelyek határoló vonalai vagy felületei végtelenül „gyűrötték” vagy „érdesekek”, illetve „szakadásosak” (szakkifejezéssel, nem differenciálhatóak)” A fraktálokat ezzel szemben bármilyen nagy nagyításban is vizsgálva, mindig találunk „gyűrődést” vagy „szakadást”, matematikai szemszögből, olyan részeket, melyeknek nem állandó, vagy nem is létezik a differenciálhányadosa. Ezek ráadásul sok esetben (a matematikai szigorúságnál jóval megengedőbb értelemben) „hasonlítanak” a teljes alakzatra vagy valamelyik kisebb nagyításban látható részletre (önhasonlóság). Éppen emiatt az olyan geometriai jellemzőik, mint a terület, terület, térfogat, ívhossz, felszín, sőt: a térdimenzió, elfajult (végtelen vagy nulla) értékeket adnak, és általában is, a térszemlélettel ellentétesen, meglepő és paradox módon viselkednek. Innen nyerték nevüket is: a latin fractus melléknév, a frangere, „törni” ige származéka, ugyanis elsősorban töröttet, darabosat (vö. „mindenütt tüskesség” vagy „mindenütt szakadásosság”), másodsorban szabálytalant, kivételest jelent. Benoit B. Mandelbrot, a fogalom névadója ebből a latin szóból alkotta meg a fraktál kifejezést. (Wikipedia)

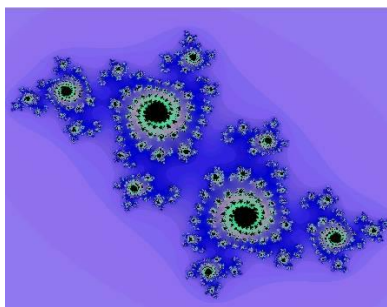
Hol találkozhatunk fraktálokkal? Fraktálokat használnak például a topológiában, a metrikában, a káosz-elmélettel kapcsolatos kutatásokban, de a természetben is fellelhetőek például az emberi erek hálózatában, a részecskék hőmozgásánál, egy partvonal tagolódásában, a felhők alakjában, egy karfiol alakjában vagy akár az értéktőzsdék áringadozásainál. Felhasználásukkal a médiában mesterséges tájakat, hegyeket, fákat is létre lehet hozni. Olyan népszerűekké váltak az utóbbi évtizedekben, hogy sokan akár hobbiból is készítenek fraktálképeket. De hogyan is készül egy-egy fraktálkép?

Fraktálkészítéshez minden esetben egy-egy rekurzív függvényre lesz szükségünk. Gyakran iterált függvényeket használunk, amelyekkel szigorúan önhasonló fraktálok hozhatók létre, mint amilyen például a Koch fraktál. Komplex függvények korlátosságának vizsgálatával bonyolultabb, összetettebb fraktálalakzatok állíthatók, ilyen például a Mandelbrot halmaz. További olvasnivalót lehet találni a <http://fraktal.lap.hu> címen.

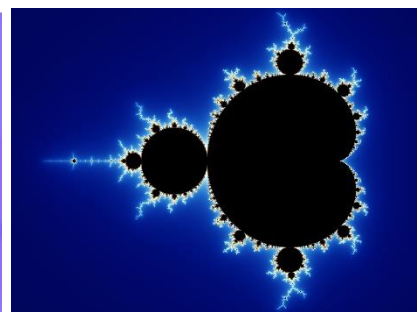
Nézzünk néhány szép fraktál alakzatot! (képek a wikipédiából származnak)



Koch farktál

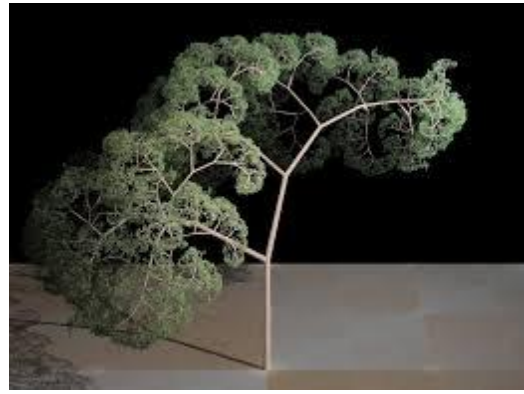


Júlia halmaz



Mandelbrot halmaz

A természetben is előfordulnak fraktál alakzatok:



Azokkal az alkalmazásokkal, amelyekkel grafika készíthető, általában fraktálrajz is előállítható – így Imagine Logo-val is! A következőkben megnézzük, hogyan állítható elő néhány érdekes fraktálalakzat. Példáinkban a korlátos, komplex függvényekkel nem foglalkozunk, a kevésbé számolásigényes feladatokra koncentrálnak.

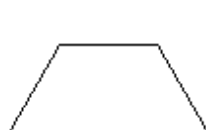
### Szakasz helyettesítése egy törött vonallal

Önhasonló alakzatokat készíthetünk úgy, hogy egy alakzat szakaszait helyettesítjük egy törött vonallal.

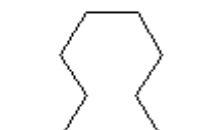
#### Sierpinski nyílhegy görbéje

Sierpinski nyílhegy görbéje úgy keletkezik, hogy egy adott hosszúságú szakaszt helyettesítünk három feleakkora hosszúságúval, az ábrának megfelelően. A második görbénél ugyanezt a módszert alkalmazzuk az első görbe szakaszaira, a harmadiknál pedig a második szakaszaira.

Készíts eljárást (nyíl :n :h) a :h hosszúságú szakaszból kiinduló :n-edik nyílhegygörbe rajzolására!



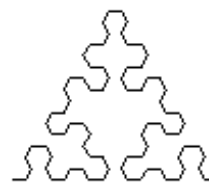
nyíl 2 100



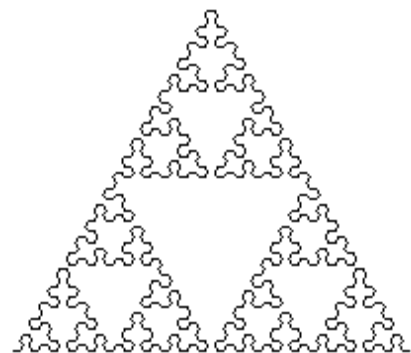
nyíl 3 100



nyíl 4 100



nyíl 5 100



nyíl 7 100

```
eljárás nyíl :n :h
```

```
  hakülönben :n=1 [előre :h]
```

```
    [balra 60 nyílj :n-1 :h/2 jobbra 60 nyíl :n-1 :h/2
```

```
     jobbra 60 nyílj :n-1 :h/2 balra 60]
```

```
vége
```

```
eljárás nyílj :n :h
```

```
  hakülönben :n=1 [előre :h]
```

```
    [jobbra 60 nyíl :n-1 :h/2 balra 60 nyílj :n-1 :h/2
```

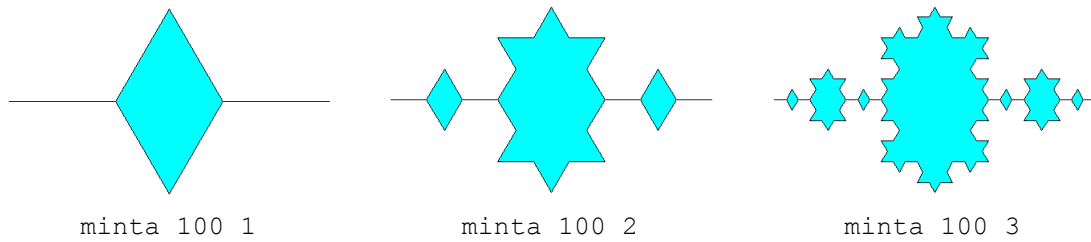
```
     balra 60 nyíl :n-1 :h/2 jobbra 60]
```

```
vége
```



### Ugyanarra a szakaszra két, a szakaszra tükrözött töröttvonalat illesztünk

Készítsd el az alábbi rekurzív ábrát rajzoló minta :h :sz eljárást, ahol :h a hossz mérete, :sz pedig a szintszám. A vonalak feketék, a belső színek kékek legyenek!



A mintában a jól ismert Koch fraktált és tükörképét ismerhetjük fel.


```
eljárás minta :sz :h
  Koch :sz :h 1 tollatfel hátra :h tollatle
  Koch :sz :h -1 tollatfel hátra :h tollatle
  tollatfel színez :sz :h tollatle
vége
```

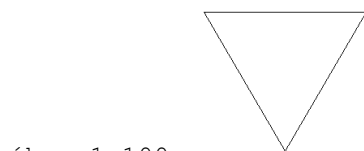
Az elkészítendő fraktált és tükörképét az :m paraméter beállításával állítjuk elő (1,-1).

```
eljárás Koch :sz :h :m
  hakülönben :sz = 0 [előre :h]
    [Koch :sz-1 :h/3 :m balra 60*:m
     Koch :sz-1 :h/3 :m jobbra 120*:m
     Koch :sz-1 :h/3 :m balra 60*:m
     Koch :sz-1 :h/3 :m]
vége

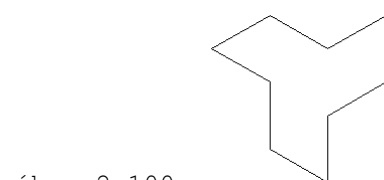
eljárás színez :sz :h
  ha :sz>0 [előre :h/2 töltőszín! 11 tölt h :h/2
    színez :sz-1 :h/3 előre :h/3*2 színez :h/3 :sz-1
    hátra :h/3*2]
vége
```

### Kiindulás nem egyetlen szakaszból, egy háromszög oldalaiból

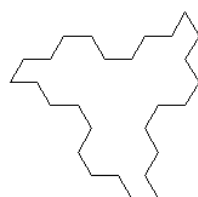
Készítsd el az alábbi rekurzív sorozatot rajzoló Logo eljárást (ábra :sorszám :hossz)! Indulj ki egy egyenlő oldalú háromszögből, minden oldalát helyettesítsd az itt látható töröttvonallal: , melynek szakaszai hossza az oldalhossz fele! Az ábra következő szintjén minden egyes vonalat helyettesítsd újra ezzel a törött vonallal, és így tovább!



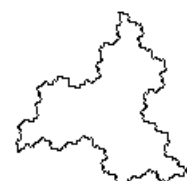
ábra 1 100



ábra 2 100



ábra 3 100



ábra 6 100

Az ábra illeszti a fraktált a háromszög oldalaira.

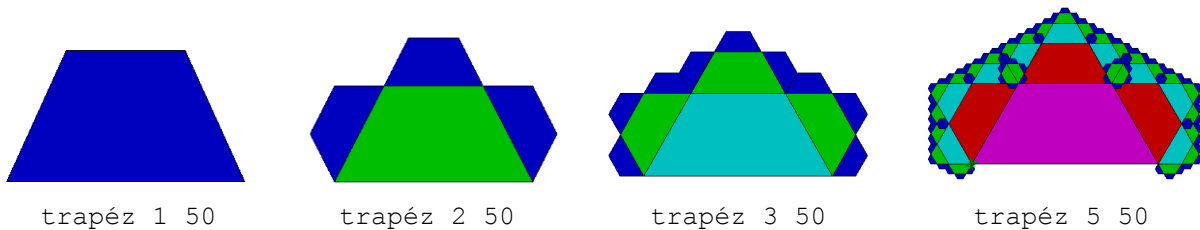
```
eljárás ábra :sorszám :hossz
  balra 30 ismétlés 3 [alap :sorszám :hossz jobbra 120] jobbra 30
vége
```

Az alap eljárás feladata, hogy egy adott szakaszt helyettesítsen egy eggyel kisebb sorszámú fraktál alakkal (töröttvonalal)

```
eljárás alap :sorszám :hossz
  hakülönben :sorszám=1 [előre :hossz]
    [balra 30 alap :sorszám-1 :hossz/2 jobbra 60 alap :sorszám-1 :hossz/2
      balra 60 alap :sorszám-1 :hossz/2 jobbra 30]
vége
```

### Nem minden szakaszra illesztünk töröttvonalat

Készíts Logo eljárást (trapéz :szint :oldal), amely színes trapézokat rajzol a trapéz három oldala :oldal, a negyedik pedig  $2 * :oldal$  hosszúságú, szögei 60, illetve 120 fokosak! Színezd ki úgy, hogy méret szerint különböző színűek legyenek a trapézok!



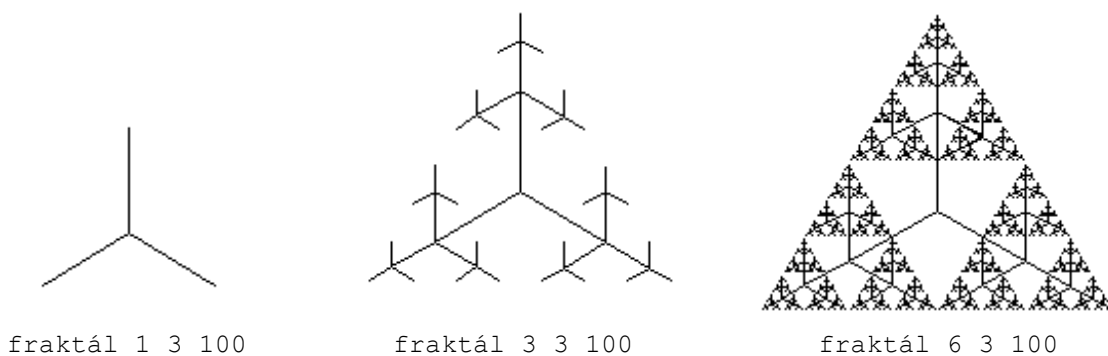
```
eljárás trapéz :szint :oldal
  ha :szint>=1 [balra 60 trapéz :szint-1 :oldal/2 jobbra 60
    előre :oldal jobbra 60 balra 60
    trapéz :szint-1 :oldal/2 jobbra 60 előre :oldal
    jobbra 60 balra 60 trapéz :szint-1 :oldal/2
    jobbra 60 előre :oldal jobbra 120
    előre 2*:oldal jobbra 120 szinez :oldal :szint]
vége
```

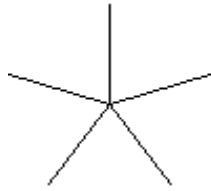
vége

```
eljárás szinez :oldal :szint
  tollatfel jobbra 30 előre :oldal töltőszín! :szint tölt
  hátra :oldal balra 30 tollatle
vége
```

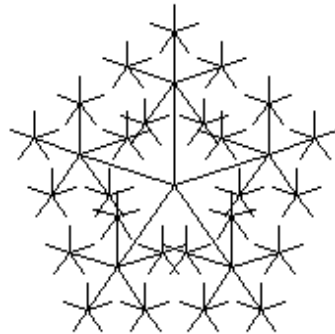
### Paraméterrel megadott számú szakaszból indulunk ki

Egy fraktál  $:m$  ágból növekszik. Az  $:n$ -edik lépésben az  $:n-1$ -edik fraktál minden ágvégéből újabb  $:m$  ág nő ki, feleakkora ághosszal. Készítsd el a fraktált rajzoló eljárást (fraktál :n :m :h), ahol :h az 1 lépésbeli ághossz!

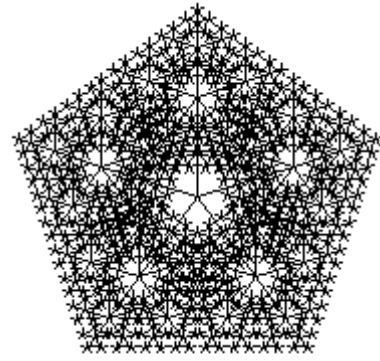




fraktál 1 5 100



fraktál 3 5 100



fraktál 6 5 100

```

eljárás fraktál :n :m :h
  ismétlés :m [előre :h ha :n>1 [fraktál :n-1 :m :h/2]
              hátra :h jobbra 360/:m]
vége

```

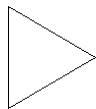
### Háromszögek felhasználása fraktál alakzatok készítésénél

Háromszögekkel érdekes rekurzív ábrákat állíthatunk össze.

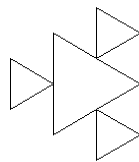
#### Az oldalak közepén újabb háromszögek nőnek ki.

Az előállítás elve legyen az, hogy minden háromszögoldal közepén egy újabb háromszög jelenjen meg.

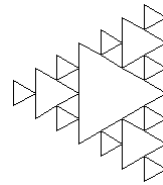
Készíts Logo eljárást (hszög :h :n), amely egy :h oldalhosszúságú háromszögből kiindulva :n-szer alkalmazza az oldalakra újabb ábrák elhelyezését!



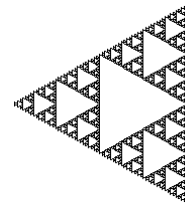
hszög 50 0



hszög 50



hszög 50 2



hszög 50 5

```

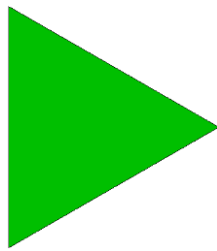
eljárás hszög :h :n
  ha :n>=0 [ismétlés 3 [előre :h/2 balra 120 hszög :h/2 :n-1
                      jobbra 120 előre :h/2 jobbra 120]]
vége

```

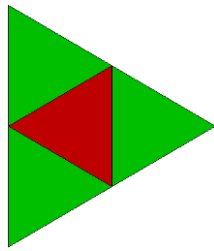
#### A háromszög közepéből kivágunk háromszögeket

A Sierpinski háromszög úgy keletkezik, hogy egy :h oldalhosszú háromszög alakú lapból kivágjuk a középső negyedrészt, majd a megmaradt három :h/2 oldalhosszú háromszögre ugyanezt alkalmazzuk. A kivágott részeket pirossal, a megmaradtakat pedig zölddel rajzoljuk.

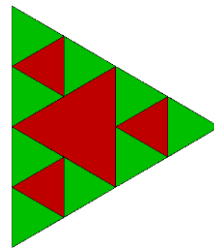
Készíts Logo eljárást (sier :db :h) a :db-edik, :h oldalhosszú Sierpinski háromszög rajzolására!



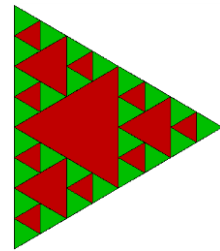
sier 0 100



sier 1 100



sier 2 100



sier 3 100

Kirajzolja a legnagyobb háromszöget és meghívja a középső háromszögeket kivágó eljárást.

```
eljárás sier :db :h
  háromszög :h 2 sierpinski :db :h
vége
```

Kivágja a középső háromszöget, majd a csúcsokhoz tartozó fele akkora szabályos háromszögeken is végrehajtja ezt a műveletet.

```
eljárás sierpinski :db :h
  előre :h/2 jobbra 60 háromszög :h/2 12 balra 60 hátra :h/2
  ha :db>0 [ismétlés 3 [sierpinski :db-1 :h/2 előre :h jobbra 120]]
vége
```

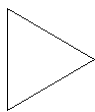
Színezett szabályos háromszög.

```
eljárás háromszög :h :szín
  töltőszín! :szín
  ismétlés 3 [előre :h jobbra 120]
  tollatfel jobbra 30 előre :h/2 tölt hátra :h/2 balra 30 tollatle
vége
```

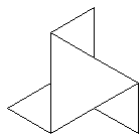
### Új háromszögek a csúcsoknál

Az előállítás elve legyen az, hogy minden háromszögoldal egyik felén egy újabb háromszög jelenjen meg!

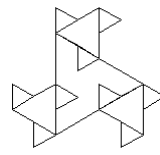
Készíts Logo eljárást (hszög :h :n), amely egy :h oldalhosszúságú háromszögből kiindulva :n-szer alkalmazza az oldalakra újabb ábrák elhelyezését!



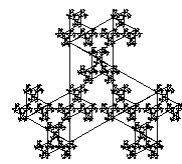
hszög 50 0



hszög 50 1



hszög 50 2



hszög 50 6

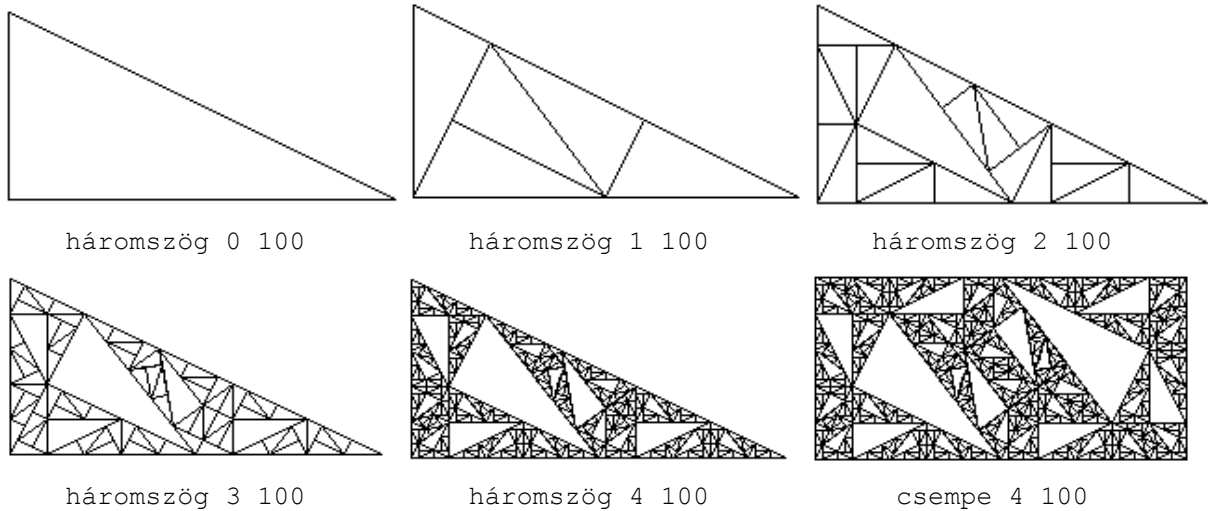
```
eljárás hszög :h :n
  ismétlés 3 [három :h :n jobbra 120]
vége
```

```
eljárás három :h :n
  különben :n=0 [előre :h]
  [előre :h/2 balra 180 három :h/2 :n-1
  jobbra 120 három :h/2 :n-1 jobbra 120
  három :h/2 :n-1 balra 60 előre :h/2]
vége
```

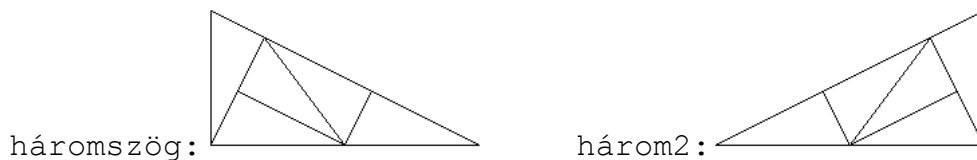
## Hasonló háromszögekre darabolt háromszög(ek)

Egy csempe (szélkerék csempe) kiindulópontja egy derékszögű háromszög, melynek egyik befogója hossza a másik befogó kétszerese. A háromszöget 5 hozzá hasonló kisebb háromszögre oszthatjuk. A felosztást a középső háromszög kivételével a maradék négy háromszögre újra és újra elvégezzük, majd a végén két ilyen sokszorosán felosztott háromszöget egymás mellé helyezve kapjuk meg egy csempe mintázatát.

Készíts Logo eljárásokat a háromszögek (háromszög :n :h) és a csempe (csempe :n :h), ahol :n a felosztások száma, :h pedig a kiinduló háromszög rövidebb befogójának hossza!



Érdekes felismerni, hogy kétféle háromszögünk van – egymás tükörképei, s mindegyik egyet tartalmaz saját magából és hármat a másiktól:



Az első ábrát megfigyelve gyorsan látható, hogy a háromszög rövidebb befogójával szemközti szöghöz tartozó tangens érték  $\frac{1}{2}$ , így ebből a szög könnyen kiszámolható az arctan függvény segítségével.

```

eljárás háromszög :n :h
  előre :h jobbra 90 + arctan 1/2 előre :h*gyök 5 jobbra 180-arctan 1/2
  előre 2*:h jobbra 90
  ha :n>0 [jobbra arctan 1/2 előre :h/gyök 5 jobbra 180
    háromszög2 :n-1 :h/gyök 5 balra 180 előre :h/gyök 5
    balra 90 háromszög2 :n-1 :h/gyök 5 hátra 2*:h/gyök 5
    balra 90 háromszög :n-1 :h/gyök 5
    háromszög2 :n-1 :h/gyök 5 előre :h/gyök 5
    jobbra 90-arctan 1/2 előre :h jobbra 90]
vége

```

A tükörcép háromszögben a fordulások iránya az ellentettje az eredetinek!

```

eljárás háromszög2 :n :h
  előre :h balra 90+arctan 1/2 előre :h*gyök 5 balra 180-arctan 1/2
  előre 2*:h balra 90
  ha :n>0 [balra arctan 1/2 előre :h/gyök 5 balra 180
    háromszög :n-1 :h/gyök 5 jobbra 180
    előre :h/gyök 5 jobbra 90 háromszög :n-1 :h/gyök 5
    hátra 2*:h/gyök 5 jobbra 90 háromszög2 :n-1 :h/gyök 5
    háromszög :n-1 :h/gyök 5 előre :h/gyök 5
    balra 90-arctan 1/2 előre :h balra 90]
vége

```

A csempe:

```

eljárás csempe :n :h
  háromszög :n :h előre :h jobbra 90 előre 2*:h jobbra 90
  háromszög :n :h balra 90 hátra 2*:h balra 90 hátra :h
vége

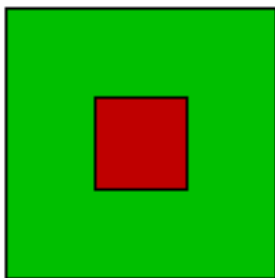
```

## Fraktálok négyzetekből építve

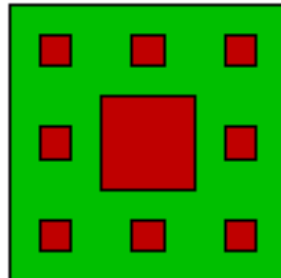
### Sierpinski négyzet, a középső négyzetet kivágjuk

A Sierpinski négyzet úgy keletkezik, hogy egy  $:h$  oldalhosszú négyzet alakú lapból kivágjuk a középső kilencedrészt, majd a megmaradt nyolc  $:h/3$  oldalhosszú négyzetre ugyanezt alkalmazzuk. A kivágott részeket pirossal, a megmaradtakat pedig zölddel rajzoljuk.

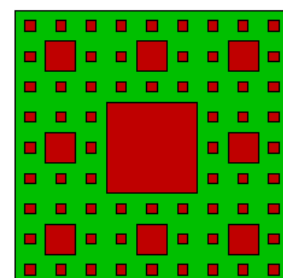
Készíts Logo eljárást (`sier :db :h`) a  $:db$ -edik,  $:h$  oldalhosszú Sierpinski négyzet rajzolására (a nulladik egy zöld négyzet)!



sier 1 100



sier 2 100



sier 3 100

Kirajzolja a legnagyobb négyzetet és kivágja a középsőket.

```

eljárás sier :db :h
  négyzet :h 2 siern :db :h
vége

```

Kivágja a középső négyzetet, majd meghívja minden oldalon kétszer ezt az eljárást.

```

eljárás siern :db :h
  ha :db>0 [tollatfel előre :h/3 jobbra 90 előre :h/3 balra 90
    tollatle négyzet :h/3 12 tollatfel jobbra 90 hátra :h/3
    balra 90 hátra :h/3 tollatle
    ismétlés 4 [ismétlés 2 [siern :db-1 :h/3 tollatfel
      előre :h/3 tollatle]
    tollatfel előre :h/3 jobbra 90 tollatle]]
vége

```

Színezett négyzet.

```

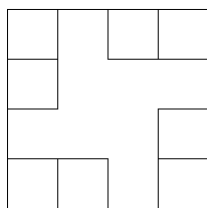
eljárás négyzet :h :szín
  töltőszín! :szín
  ismétlés 4 [előre :h jobbra 90]
  tollatfel jobbra 45 előre :h/2 tölt hátra :h/2 balra 45 tollatle
vége

```

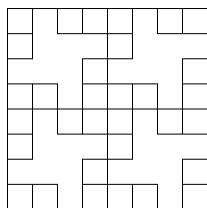
### Darabolás négyzetekkel

Egy rekurzív ábrát úgy készítünk, hogy egy négyzetet felosztunk 4\*4 kisebb négyzetre, az így kapott négyzeteket újra 4\*4 kisebb négyzetre, és így tovább. A legutolsó felosztásnál azonban a belső négyzetek közül csak az ábrán látható 8 négyzet határvonalait rajzoljuk ki.

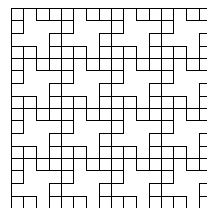
Készíts Logo programot (négyzet :db :hossz), amely a :hossz méretű négyzet felosztását :db-szor végzi el!



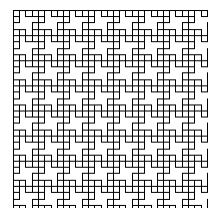
négyzet 1 100



négyzet 2 100



négyzet 3 100



négyzet 4 100

```

eljárás négyzet :db :hossz
  ismétlés 4 [rekurzív :db :hossz jobbra 90]
vége

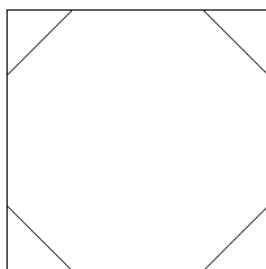
eljárás rekurzív :db :hossz
  hakülönben :db=0 [előre :hossz]
  [ismétlés 2 [ismétlés 4 [rekurzív :db-1 :hossz/2 jobbra 90]
    előre :hossz]]

```

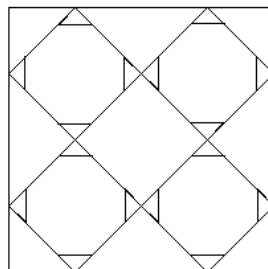
vége

### Darabolás, csücskök levágása

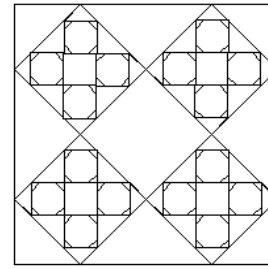
Készítsd el a következő rekurzív ábrát (négyzetek :szint :oldal) kirajzoló programot, amely egy négyzet csücskeiből rendre levág egy-egy darabot, majd ezekre újabb négyzeteket rajzol!



négyzetek 1 200



négyzetek 2 200



négyzetek 3 200

A négyzetek sarkaiban négyszer hívjuk meg a rekurzív ábra rajzolását.

```

eljárás négyzetek :szint :oldal
  ha :szint>0 [ismétlés 4 [előre :oldal
    csücsök :szint-1 :oldal/4
    jobbra 90]]

```

vége

Minden sarokban meghívjuk az eredeti ábrához hasonló alakzat kirajzolását.

```

eljárás csücsök :szint :oldal
  hátra :oldal jobbra 45
  négyzetek :szint :oldal*gyök 2
  előre :oldal*gyök 2 hátra :oldal*gyök 2 balra 45 előre :oldal
vége

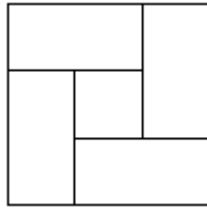
```

### Darabolás téglalapokkal

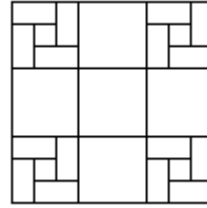
Az alábbi ábrát egy rekurzív Logo eljárás rajtolta, melynek első paramétere 0, 1, 2, illetve 3, a második paramétere pedig a nagy négyzet oldalhossza volt. Készíts rá Logo eljárást!



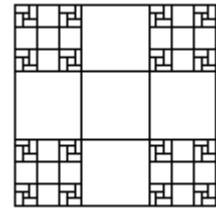
fel 0 100



fel 1 100



fel 2 100



fel 3 100

```

eljárás fel :n :hossz
  hakülönben :n=0 [ismétlés 4 [előre :hossz jobbra 90]]
  [ismétlés 4 [tégla :n :hossz/3 előre :hossz jobbra 90]]
vége

```

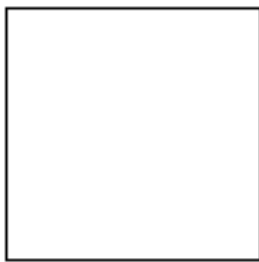
```

eljárás tégla :n :hossz
  ismétlés 2 [előre 2*:hossz jobbra 90 előre :hossz jobbra 90]
  ha :n>1 [fel1 :n-1 :hossz]
vége

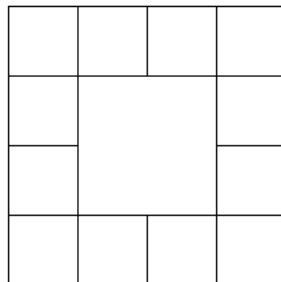
```

### Keretes darabolás

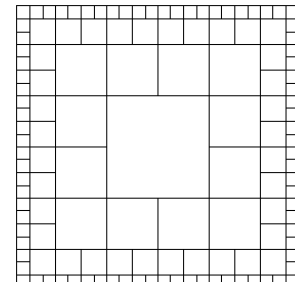
Készíts Logo eljárást (négy :n :h), amely négyszögeket rajzol egymás körül! Legyen :n darab négyzetsor egymás körül, a legbelső négyzet oldalhossza legyen :h, kifelé haladva a négyzetek oldalhossza feleződjön!



négy 1 100



négy 2 100



négy 4 100

```

eljárás négy :db :h
  négyzetsorok :db 1 :h
vége

```

Kirajzolja a :db négyzetsort, négyzetekből álló keretet.

```

eljárás négyzetsorok :db :n :h
  négyzetsor :n :h
  ha :db>1 [hátra :h/2 jobbra 90 hátra :h/2 balra 90
  négyzetsorok :db-1 :n*2+2 :h/2]
vége

```



Oldalanként : n négyzettel kirajzol egy keretet.

```

eljárás négyzetsor :n :h
  ismétlés 4 [ismétlés :n [négyzet :h előre :h] jobbra 90]
vége
  
```

```

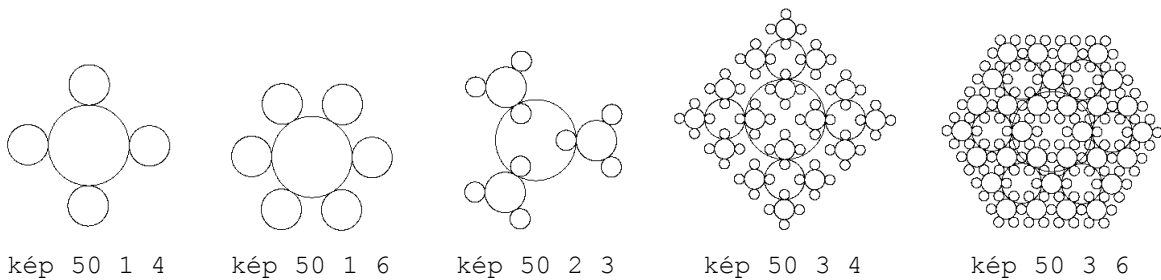
eljárás négyzet :h
  ismétlés 4 [előre :h jobbra 90]
vége
  
```

## Fraktálok körökből építve

### Körvonalon elhelyezkedő újabb körök

Körökből úgy állítunk elő rekurzív ábrát, hogy a körvonal mentén :db darab újabb ábrát helyezünk el fele akkora méretben, ezekre újabb ábrákat illesztünk, és így tovább, mindezt :n-szer ismételve.

Készíts Logo eljárást (kép :r :n :db), amely egy :r sugarú körből kiindulva végzi el ezt!



kép 50 1 4

kép 50 1 6

kép 50 2 3

kép 50 3 4

kép 50 3 6

```

eljárás kép :r :n :db
  körtrajzol :r
  ha :n>0 [ismétlés :db [ismétlés 360/:db [előre 3,14159*:r/180 balra 1]
    jobbra 180 kép :r/2 :n-1 :db jobbra 180]]
vége
  
```

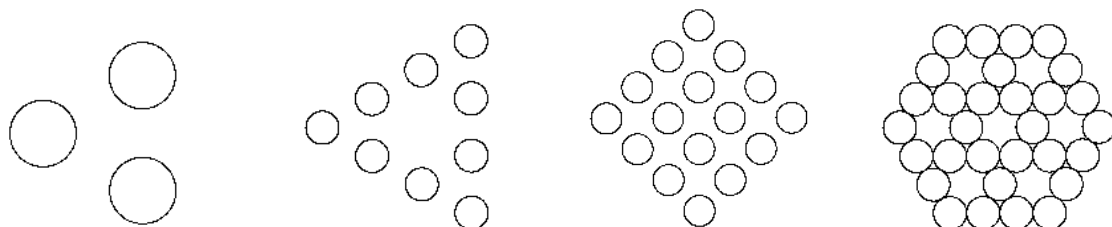
Kör rajzolása :r sugárral:

```

eljárás körtrajzol :r
  ismétlés 360 [előre 3,14159*:r/180 balra 1]
vége
  
```

### Nem minden fraktálszint látszódik

Az erdei gombák gyakran ún. boszorkánykörök mentén találhatók. Készítsd el a gomba :év :db :r eljárást! Az első évben egyetlen gomba nő. A következő évben :db darab gomba fejlődik egy :r sugarú kör vonala mentén szabályos eloszlásban. A következő években mindig az előző gombák körüli kör mentén lesznek gombák. A körök sugara és a gombák mérete mindig az előző évi sugár felére csökkenjen!



gomba 2 3 50

gomba 3 3 50

gomba 3 4 50

gomba 3 6 50

```

eljárás gomba :év :db :r
  tollatfel balra 90 előre :r jobbra 90
  hakülönben :év=1 [tollatle körív :r 360 tollatfel]
                    [ismétlés :db [gomba :év-1 :db :r/2 tollatfel
                                   körív :r 360/:db]]
  tollatfel balra 90 hátra :r jobbra 90 tollatle
vége

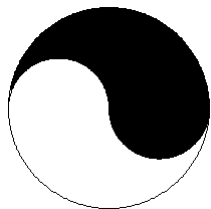
eljárás körív :r :fok
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége

```

### Körben körívek – Jinjang

A kínai *jinjang* szimbólum egy kör, amit egy két félkörvonal oszt két részre, az egyik fekete, a másik része pedig fehér színű.

Készíts Logo eljárást (*jinjang :r :db*), amely rekurzívan, a *jinjang* szimbólumba írt feleak-kora kör alakú területekre az ábrának megfelelően *:db* mélységig újra alkalmazza az eljárást!



jinjang 100 1



jinjang 100 2



jinjang 100 3

```

eljárás jinjang :r :db
  töltőszín! 0
  ismétlés 360 [előre :r*3,14159/180 jobbra 1]
  fraktál :r/2 :db
vége

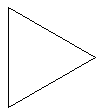
eljárás fraktál :r :db
  tollatfel jobbra 90 előre 3*:r/2 jobbra 90 tollatle
  ha :db>1 [fraktál :r/4 :db-1]
  tollatfel balra 90 előre 2*:r jobbra 90 tollatle
  ha :db>1 [fraktál :r/4 :db-1]
  tollatfel balra 90 hátra 7*:r/2 balra 90 tollatle
  ismétlés 360 [előre :r*3,14159/90 jobbra 1]
  ismétlés 180 [előre :r*3,14159/180 jobbra 1]
  ismétlés 180 [előre :r*3,14159/180 balra 1]
  ismétlés 90 [előre :r*3,14159/90 balra 1]
  tollatfel balra 90 előre 2 töltőszín! 0 tölt hátra 2 jobbra 90
  tollatle
  ismétlés 90 [előre :r*3,14159/90 balra 1] jobbra 180
vége

```

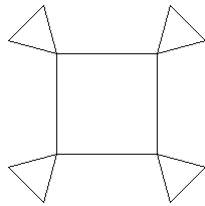
## Fraktálkészítés többféle sokszög felhasználásával

### A sokszög sarkaiban kifelé rajzolva újabb sokszögek

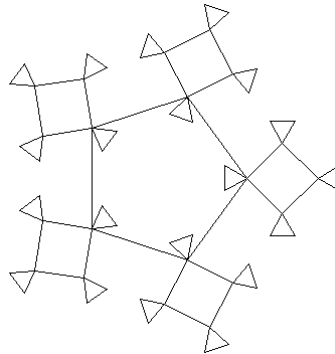
Készítsd el a következő rekurzív ábra rajzolóprogramját (ábra :n :h), ahol :n oldalú szabályos sokszöget rajzolunk, majd a sarkain egyre kisebb oldalszámúakat, feleakkora oldalhosszal!



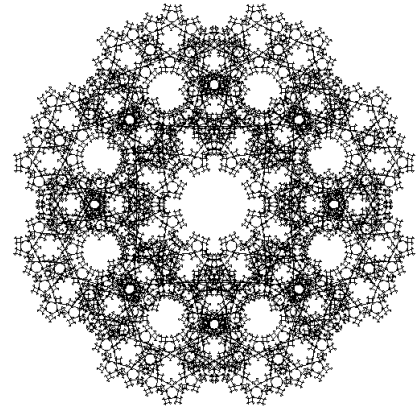
ábra 3 80



ábra 4 80



ábra 5 80



ábra 8 80

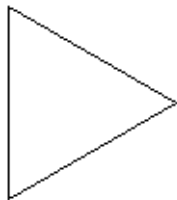
eljárás ábra :n :h

```
ismétlés :n [előre :h jobbra 180/:n+180/(:n-1)-180
             ha :n>3 [ábra :n-1 :h/2]
             balra 180/:n+180/(:n-1)-180 jobbra 360/:n]
```

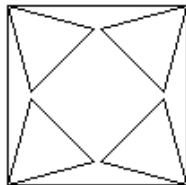
vége

### A sokszög sarkaiban befelé rajzolva újabb sokszögek

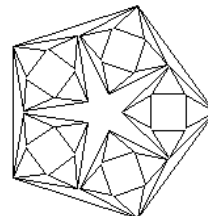
Készítsd el a következő rekurzív ábra rajzolóprogramját (ábra :db :hossz), ahol :db oldalú szabályos sokszöget rajzolunk, majd a sarkain egyre kisebb oldalszámúakat, feleakkora oldalhosszal!



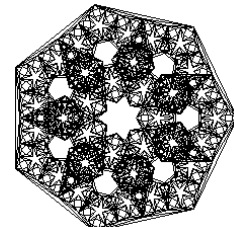
ábra 3 80



ábra 4 80



ábra 5 80



ábra 7 80

Az elfordulások szögét a :db és :db-1 csúcsú sokszögek szögeinek különbségéből lehet kiszámolni.

eljárás ábra :db :hossz

```
ha :db>2
  [ismétlés :db
   [előre :hossz jobbra 360/:db-(360/:db-360/(:db-1))/2
   ábra :db-1 :hossz/2 jobbra (360/:db-360/(:db-1))/2]]
```

vége

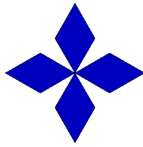
## Vezérlő eljárással készülő fraktálok

### Jégvirág

Egy jégvirág a következőképpen növekszik. Az első időegységben négy rombusz alakú levélből áll. A következő időegységben a rombuszok külső csúcsából kinő három-három újabb rombusz alakú

levél, egymással és a nagyobb rombuszsal 90 fokos szöget bezárva. A következő időegységben ezek külső csúcsából újra két levél nő ki, ... és így tovább.

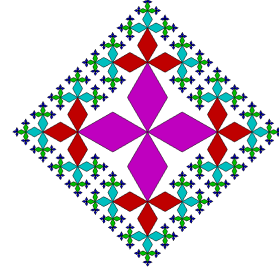
Készíts Logo eljárást (jégvirág :idő :hossz), amely kirajzolja a növény :idő időegység-beli állapotát! A rombusz oldalhossza legyen :hossz, belső szögei pedig 60, illetve 120 fokosak!



jégvirág 1 30



jégvirág 2 30



jégvirág 5 30

A jégvirág 4-es szimmetriájú, ugyanakkor az egyes szirmok 3 fele ágaznak el, ezért használunk egy „vezérlő” eljárást, amely a rekurzív rajzolót meghívja négyszer.

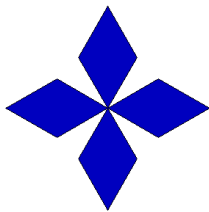
```
eljárás jégvirág :idő :hossz
  ismétlés 4 [frakt :idő :hossz jobbra 90]
vége
```

```
eljárás frakt :idő :hossz
  balra 30 előre :hossz jobbra 60 előre :hossz balra 30
  ha :idő>1 [balra 90 ismétlés 3 [frakt :idő-1 :hossz/2 balra 90]
    jobbra 180]
  jobbra 150 előre :hossz jobbra 60 előre :hossz balra 30
  tollatfel hátra :hossz töltőszín! :idő tölt előre :hossz tollatle
vége
```

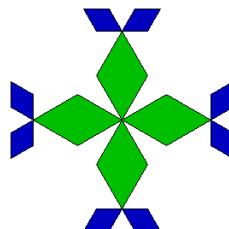
## Növény

Egy növény a következőképpen növekszik. Az első időegységben négy rombusz alakú levélből áll. A következő időegységben a rombuszok külső csúcsából kinő két-két újabb rombusz alakú levél, egymással és a nagyobb rombuszsal 120 fokos szöget bezárva. A következő időegységben ezek külső csúcsából újra két levél nő ki, ... és így tovább.

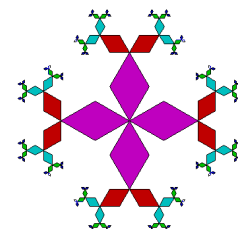
Készíts Logo eljárást (növény :idő :hossz), amely kirajzolja a növény :idő időegység-beli állapotát! A rombusz oldalhossza legyen :hossz, belső szögei pedig 60, illetve 120 fokosak!



növény 1 30



növény 2 30



növény 5 30

Szükség van egy „vezérlő” eljárásra, amelyik meghívja négyszer a fraktál ágakat:

```
eljárás növény :idő :hossz
  ismétlés 4 [növényág :idő :hossz jobbra 90]
vége
```

```

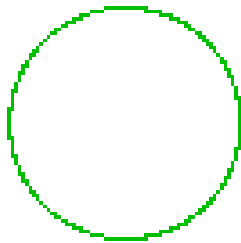
eljárás növényág :idő :hossz
  balra 30 előre :hossz jobbra 60 előre :hossz balra 30
  ha :idő>1 [balra 60 ismétlés 2 [növényág :idő-1 :hossz/2 jobbra 120]
    balra 180]
  jobbra 150 előre :hossz jobbra 60 előre :hossz balra 30 tollatfel
  hátra :hossz/2 töltőszín! :idő tölt előre :hossz/2 tollatle jobbra 180
vége

```

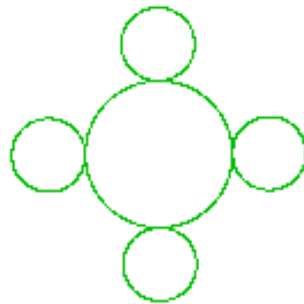
## Bimbó

Egy bimbós növény a 0. lépésben egyetlen kör alakú bimbóból áll. A következő lépésben megjelenik rajta 4, feleakkora bimbó, a következő időegységben azokon három, megint feleakkora bimbó, s így tovább, a bimbók száma egyre kevesebb, mérete egyre kisebb lesz.

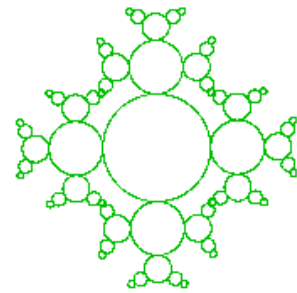
Készíts Logo eljárást (bimbó :n :h), amely az :n-edik időegységben rajzolja a bimbós növényt, :h mérettel!



bimbó 0 50



bimbó 1 50



bimbó 4 50

A bimbó :év :méret a „középső kör” kirajzolását végezi el és meghívja negyedkörönként a kisebb bimbók kirajzolását, amelyet a frakt :év :méret :db eljárás meghívásával érünk el. Az első szinten 4, a többin csak 3 rekurzív hívásra van szükség, ezért az első szint eljárást külön választjuk.

```

eljárás bimbó :év :méret
  tollatfel előre :méret balra 90 tollatle tollszín! 2
  ismétlés 4 [körív -:méret 90 ha :év>0 [frakt :év :méret/2 3]]
  tollatfel jobbra 90 hátra :méret tollatle
vége

```

```

eljárás frakt :év :méret :db
  ismétlés :db [körív :méret 360/(:db+1)
    ha :év>1 [balra 180 frakt :év-1 :méret/2 :db-1
      jobbra 180]]
  körív :méret 360/(:db+1)
vége

```

```

eljárás körív :r :fok
  ismétlés :fok [előre :r*3,14159/180 jobbra 1]
vége

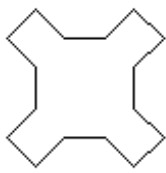
```

## Peano görbe

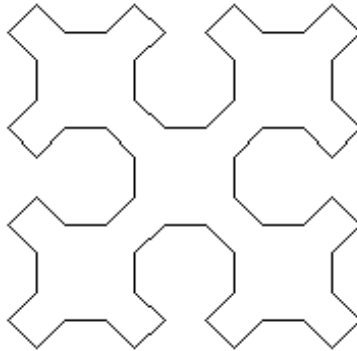
Az egyik Peano görbe egy négyzet 4 csúcsára helyezett négyzetekből áll. A második lépésben minden egyes négyzetet egy-egy újabb Peano görbe helyettesít, ... és így tovább.

Készíts eljárást peano :n :h az :n-edik Peano görbe rajzolására, amelynek az oldalai :h hosszúságúak!

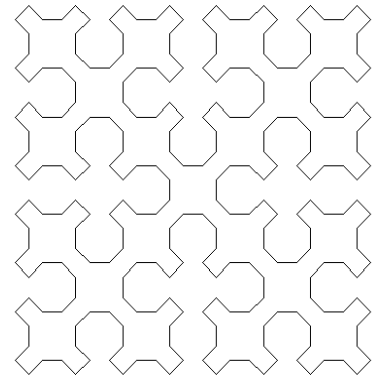
A Peano görbének négyes szimmetriája van, azaz ha az alsó sorban látható eredményű p eljárást megírod, akkor a Peano görbe mindig 4 ilyen egymáshoz illesztéséből áll.



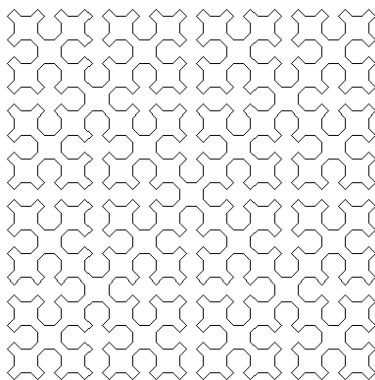
peano 1 20



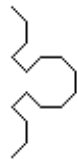
peano 2 20



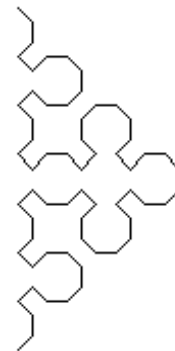
peano 3 10



peano 4 5



p 2 20



p 3 20

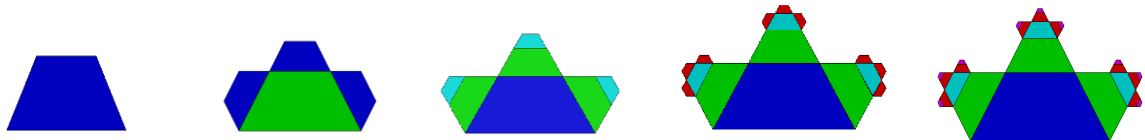
```
eljárás peano :n :h
  ismétlés 4 [p :n :h jobbra 45 előre :h jobbra 45]
vége
```

```
eljárás p :n :h
  hakülönben :n=1 [jobbra 45 előre :h balra 45 előre :h balra 45
    előre :h jobbra 45]
  [p :n-1 :h jobbra 45 előre :h jobbra 45 p :n-1 :h
    balra 90 előre :h balra 90 p :n-1 :h jobbra 45
    előre :h jobbra 45 p :n-1 :h]
vége
```

## Szintenként másként viselkedő fraktálok

### Trapézok, páros-páratlan vizsgálattal

Készíts Logo eljárást (trapéz :szint :oldal :hányadik), amely színes trapézokat rajzol (a trapéz három oldala :oldal, a negyedik pedig  $2 * :oldal$  hosszúságú, szögei 60, illetve 120 fokosak) váltakozva vagy csak egy vagy mind a három :oldal hosszúságú oldalra. A :hányadik paraméteren keresztül adjuk meg, hogy egy vagy három trapézt kell-e rajzolni. Színezd ki úgy, hogy méret szerint különböző színűek legyenek a trapézok!



trapéz 1 50

trapéz 2 50

trapéz 3 50

trapéz 4 50

trapéz 5 50

eljárás trapéz :szint :oldal :hányadik

ha :szint >= 1

[ha maradék :hányadik 2=1

[balra 60 trapéz :szint-1 :oldal/2 :hányadik+1 j 60]

előre :oldal jobbra 60 balra 60

trapéz :szint-1 :oldal/2 :hányadik+1

jobbra 60 előre :oldal jobbra 60

ha maradék :hányadik 2=1

[balra 60 trapéz :szint-1 :oldal/2 :hányadik+1 jobbra 60]

előre :oldal jobbra 120 előre  $2 * :oldal$  jobbra 120

színez :oldal :hányadik]

vége

eljárás színez :oldal :szint

tollatfel jobbra 30 előre :oldal töltőszín! :szint tölt

hátra :oldal balra 30 tollatle

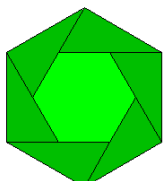
vége

### Hatszög, egymást hívó rekurzív eljárásokkal

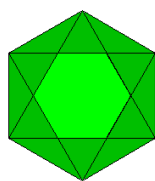
Egy hatszög belsejébe kétféle (A, illetve B típusú) összekötő vonalakkal helyezhetünk el egy másik hatszöget. A belső hatszögbe újabb hatszöget tehetünk, s abba pedig természetesen még újabbakat. Az összekötő vonalak befelé haladva mindig változnak (A típusú belsejében B típusú, B típusú belsejében pedig A típusú van). Egy hatszög oldalhossza mindig az öt közvetlenül tartalmazó hatszög oldalhosszának gyök(3)-ad része.

A hatszögeket színeznünk is kell! A külső és a hozzá legközelebb levő hatszög között sötétzöldek legyenek a háromszögek; eggyel beljebb világoszöldek, még eggyel beljebb újra sötétzöldek, ... és így tovább! A legbelső hatszög belsejét is színeznünk kell!

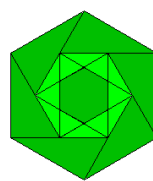
Készítsd el a két hatszög rajzoló eljárást (hata :db :h :szín, hatb :db :h :szín)! A külső hatszögben :db darab hatszögnek kell elhelyezkedni, :h legyen a külső hatszög oldalhossza, :szín pedig a külső háromszögek színe!



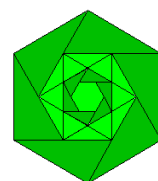
hata 1 100 2



hatb 1 100 2



hata 2 100 2



hata 3 100 2

Egymást kölcsönösen hívó két rekurzív eljárást kell készíteni.

```

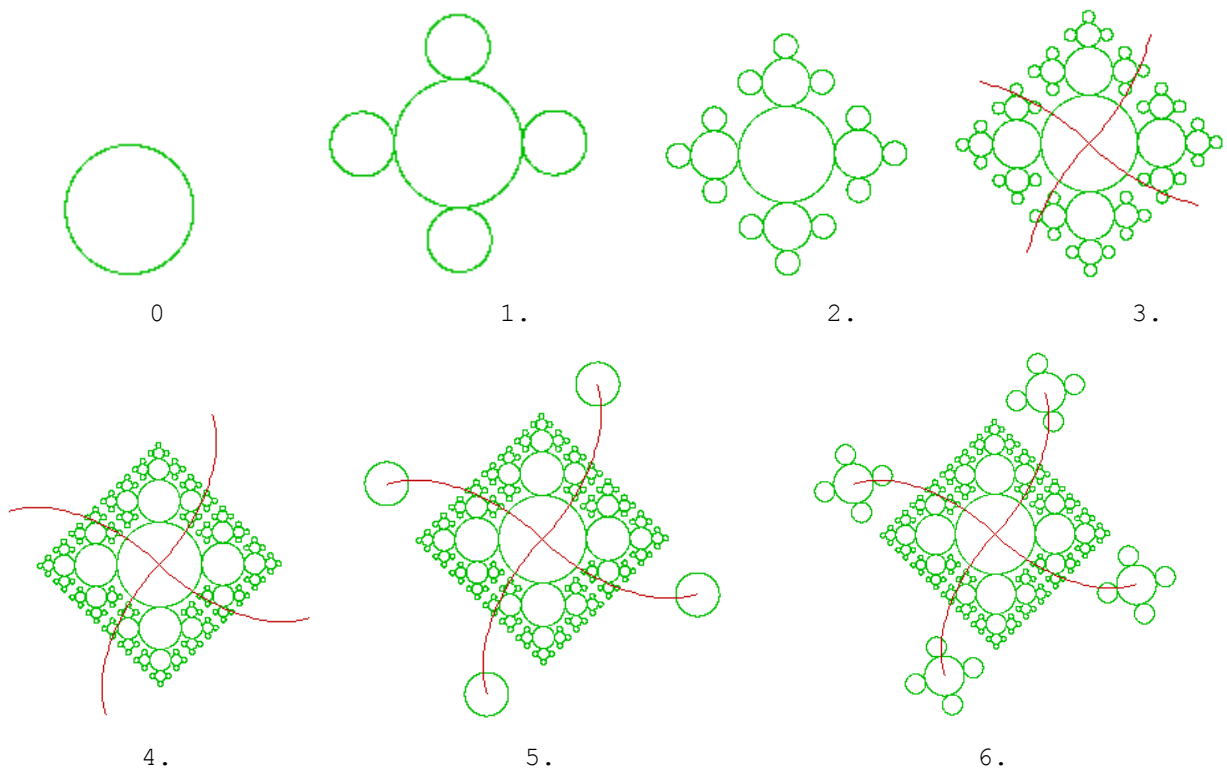
eljárás hata :db :h :szín
  ismétlés 6 [jobbra 90 előre :h/3*gyök 3 hátra :h/3*gyök 3
             balra 90 előre :h jobbra 60]
  jobbra 90 előre :h/3*gyök 3
  ismétlés 6 [előre :h/3*gyök 3 balra 30 tollatfel hátra :h/3
             töltőszín! :szín tölt előre :h/3 tollatle
             balra 30]
  hakülönben :db=1 [tollatfel balra 30 előre :h/3
                   töltőszín! 12-:szín tölt hátra :h/3 jobbra 30
                   tollatle]
                   [balra 120 hatb :db-1 :h/3*gyök 3 12-:szín jobbra 120]
  hátra :h/3*gyök 3 balra 90
vége
  
```

```

eljárás hatb :db :h :szín
  ismétlés 6 [jobbra 90 előre :h*gyök 3 hátra :h*gyök 3
             balra 90 előre :h jobbra 60]
  jobbra 90 előre :h/3*gyök 3
  ismétlés 6 [előre :h/3*gyök 3 jobbra 60 tollatfel
             előre :h/6 töltőszín! :szín tölt hátra :h/6
             jobbra 90 előre :h/6 tölt hátra :h/6
             jobbra 150 tollatle]
  hakülönben :db=1 [tollatfel balra 30 előre :h/3 töltőszín! 12-:szín
                   tölt hátra :h/3 jobbra 30 tollatle]
                   [balra 120 hata :db-1 :h/3*gyök 3 12-:szín jobbra 120]
  hátra :h/3*gyök 3 balra 90
vége
  
```

### Növény, újabb fraktálok megjelenése

Egy növény az alábbi szabályok szerint változik:





A 0. lépésben egy kör alakú részből áll, amin később újabb és újabb kör alakú részek jelennek meg. A 3. lépésben kinő belőle 4 hajtás. A hajtások körívek legyenek! Az 5. lépésben megjelenő négy új kör pontosan úgy viselkedik, mint a 0. lépésben megjelent kör, csak fele akkora méretű, továbbá a belső ábra ettől a lépéstől kezdve már nem változik tovább.

Készíts Logo eljárást (növény :n :r) a rekurzív növény :n-edik állapotának rajzolására, ahol :r a mérettel arányos!

A feladat megoldásához szükség van egy „vezérlő” eljárásra (növény :n :r), amely negyedkörönként meghívja a fraktáldrajzoló eljárást (frakt :n :r) és a hajtás :n :m :r :h eljárást. Figyeljük meg, hogy a hajtás eljárás meghívja a növény eljárást is (közvetett rekurzió), hiszen a hajtások végén újabb kis növények nőnek.

```
eljárás növény :n :r
  tollatfel előre :r balra 90 tollatle tollszín! 2
  ismétlés 4 [körív -:r 90 ha :n>0 [hakülönben :n>4 [frakt 4 :r/2]
                                                    [frakt :n :r/2]]]
  tollatfel jobbra 90 hátra :r tollatle tollszín! 4
  ha :n>2 [jobbra 60
          ismétlés 4
            [hakülönben kisebb? :n-2 2 [hajtás :n-3 :n-2 :r/2 10]
          [hajtás :n-3 2 :r/2 10]
          jobbra 90]
          balra 60]
vége

eljárás frakt :n :r
  ismétlés 3 [körív :r 90 ha :n>1 [balra 180 frakt :n-1 :r/2 jobbra 180]]
  körív :r 90
vége

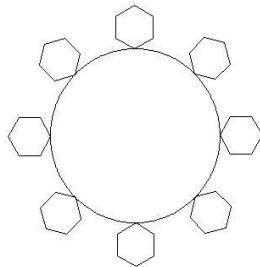
eljárás hajtás :n :m :r :h
  bkörív :h*:r 30
  hakülönben :m>1 [hajtás :n-1 :m-1 :r :h/2]
                  [ha :n>0 [növény :n-1 :r]]
  jobbra 180 körív :h*:r 30 balra 180
vége

eljárás körív :r :szög
  ismétlés :szög [előre 2*:r*3,14159/360 jobbra 1]
vége

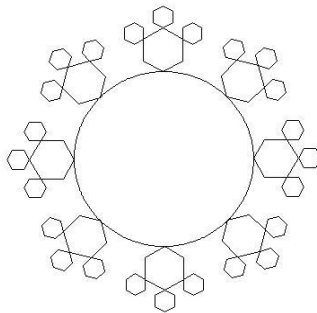
eljárás bkörív :r :szög
  ismétlés :szög [előre 2*:r*3,14159/360 balra 1]
vége
```

## Csipke

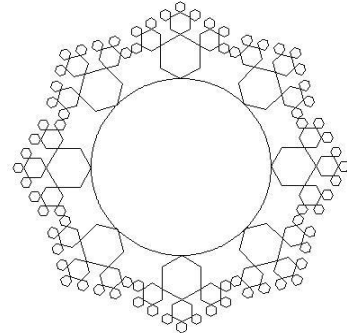
Készítsd el a következő csipke :r :n eljárást, amely egy nyolcas szimmetriájú csipke terítőt rajzol, amely :r sorból áll és az egyes csipkék mérete :n! A csipke darabok hatszögekből állnak, és a következő sor, csak a hatszög „külső” három csúcsára illeszkedik.



csipke 100 1



csipke 100 2



csipke 100 3

Egy körvonalból nyolc hatszög alapra épülő elem nő ki:

```
eljárás csipke :r :n
  ismétlés 8 [ismétlés 360/8 [előre 2*: $r$ *3,14159/360 jobbra 1]
    balra 150 hatszög :n : $r$ /4 jobbra 150]
```

vége

A hatszög alakú elemekből rekurzívan újabb 3-3 hatszög nő ki:

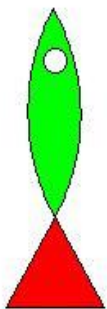
```
eljárás hatszög :n :oldal
  ha : $n$ >0 [előre :oldal jobbra 60
    ismétlés 3 [előre :oldal balra 120
      hatszög : $n$ -1 :oldal/2 jobbra 180]
    ismétlés 2 [előre :oldal jobbra 60]]
```

vége

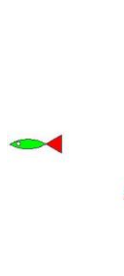
## Rajzok, mint a fraktálrajz elemei

### Halak, 1. verzió

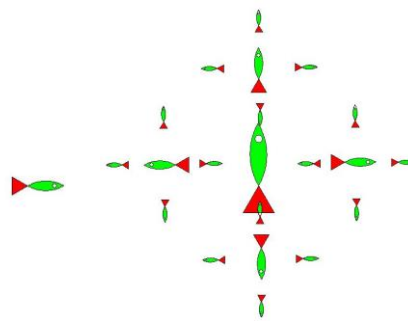
Készítsd el azokat az eljárásokat, amelyek egy színes halat (hal :méret), illetve egy halrajt rajzol ki (halraj :méret :szint :db). A középső hal körül :db nála kisebb hal úszik, akik körül újabb ezeknél is kisebb halacska van. A :db adja meg, hogy hány hal van a középső körül, a :szint pedig azt jelenti, hogy milyen mélységben kell elképzelni ezt..



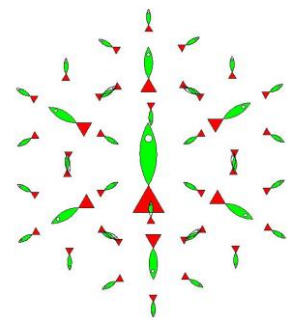
hal 100



halraj 100 2 4



halraj 100 3 4



halraj 100 3 6

A megoldás érdekessége, hogy a rekurzív hívást egy ciklusban alkalmazzuk!

```

eljárás halraj :méret :szint :db
  ha :szint>0 [tollatle hal :méret tollatfel
              ismétlés :db [előre :méret*1,5
                            halraj :méret/2 :szint-1 :db
                            hátra :méret*1,5 jobbra 360/:db]]
vége

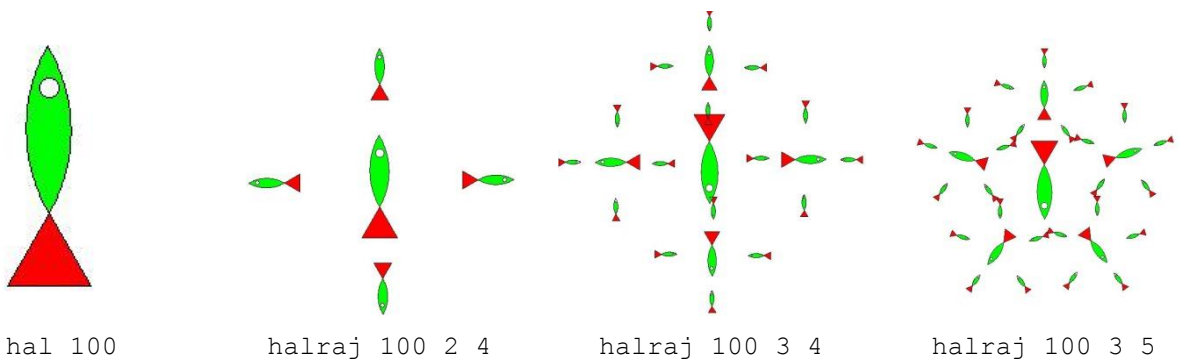
eljárás hal :méret
  tollatfel hátra :méret/4 balra 30 tollatle
  ismétlés 2 [ív :méret jobbra 120]
  hátra :méret/2 előre :méret/2 jobbra 60 hátra :méret/2
  jobbra 60 előre :méret/2
  hátra :méret/2 balra 60 előre :méret/2 balra 60
  jobbra 30 tollatfel előre 3*:méret /4 tollatle kör :méret/8
  tollatfel hátra 3*:méret/4 tollatle balra 30
  tollatfel jobbra 30 előre :méret/3 töltőszín! "zöld tölt
  hátra :méret/3 balra 180 előre :méret/4
  töltőszín! "vörös tölt
  hátra :méret/3 jobbra 150 jobbra 30 előre :méret/4 tollatle
vége

eljárás ív :méret
  ismétlés 60 [előre :méret*2*3,14159/360 jobbra 1]
vége

```

### Halak, 2. verzió

Készítsd el azokat az eljárásokat, amelyek egy színes halat (hal :méret) illetve egy halrajt rajzol ki (halraj :méret :szint :db). A középső hal körül :db nála kisebb hal úszik, akik körül újabb, ezeknél is kisebb halacska van. Figyeld meg, hogy a halak iránya szintenként megváltozik! Egyszer a nagyobb hal felé fordulnak, máskor pedig éppen ellenkezőleg, attól elfordulnak. A :db adja meg, hogy hány hal van a középső körül, a :szint pedig azt jelenti, hogy milyen mélységben kell elképzelni ezt.



A páratlan szinteken a hal másik irányban áll. Ne felejtsük el a rajzolás után visszafordítani az eredeti irányba!

```

eljárás halraj :méret :szint :db
  ha :szint>0 [tollatle
              ha maradék :szint 2 = 1 [jobbra 180]
              hal :méret
              ha maradék :szint 2 = 1 [jobbra 180]
              ismétlés :db [tollatfel előre :méret*1,5 tollatle
                            halraj :méret/2 :szint-1 :db
                            tollatfel hátra :méret*1,5
                            jobbra 360/:db tollatle]]
vége

```

```

eljárás hal :méret
  tollatfel hátra :méret/4 balra 30 tollatle
  ismétlés 2 [ív :méret jobbra 120]
  hátra :méret/2 előre :méret/2 jobbra 60 hátra :méret/2
  jobbra 60 előre :méret/2
  hátra :méret/2 balra 60 előre :méret/2 balra 60
  jobbra 30 tollatfel előre 3*:méret/4 tollatle kör :méret/8
  tollatfel hátra 3*:méret/4 tollatle balra 30
  tollatfel jobbra 30 előre :méret/3 töltőszín! "zöld tölt
  hátra :méret/3 balra 180 előre :méret/4
  töltőszín! "vörös tölt
  hátra :méret/3 jobbra 150 jobbra 30 előre :méret/4 tollatle
vége

```

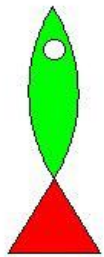
```

eljárás ív :méret
  ismétlés 60 [előre :méret*2*3,14159/360 jobbra 1]
vége

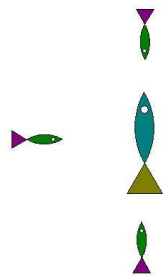
```

### Halak, 3. verzió

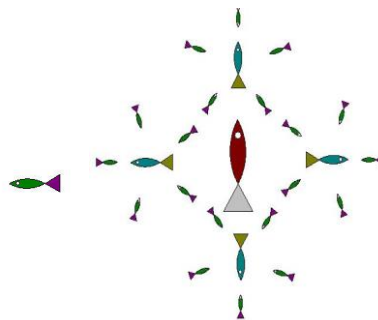
Készítsd el azokat az eljárásokat, amelyek egy színes halat (hal :méret :szín1 :szín2) illetve egy halrajt rajzol ki (halraj :méret :szint :db)! A középső hal körül :db nála kisebb hal úszik, akik körül újabb, ezeknél is kisebb halacska van. Figyeld meg, hogy a halak iránya szintenként megváltozik! Egyszer a nagyobb hal felé fordulnak, máskor pedig éppen ellenkezőleg, attól elfordulnak. A :db adja meg, hogy hány hal van a középső körül, a :szint pedig azt jelenti, hogy milyen mélységben kell elképzelni ezt. Szintenként a kiinduló :db érték eggyel nő.



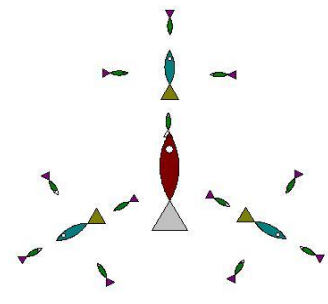
hal 100 2 4



halraj 100 2 4



halraj 100 3 4



halraj 100 3 3

```

eljárás hal :méret :szin :szin2
  tollatfel hátra :méret/balra 30 tollatle
  ismétlés 2 [ív :méret jobbra 120]
  ;farok
  hátra :méret/2 előre :méret/2
  jobbra 60 hátra :méret /2 jobbra 60 előre :méret/2
  hátra :méret/2 balra 60 előre :méret/2 balra 60
  ;szem
  jobbra 30 tollatfel előre 3*:méret/4 tollatle
  kör :méret/8
  tollatfel hátra 3*:méret/4 tollatle balra 30
  ;színez
  tollatfel jobbra 30 előre :méret/3 töltőszín! :szin tölt
  hátra :méret/3 balra 180 előre :méret/4 töltőszín! :szin2
  tölt hátra :méret/3 jobbra 150 jobbra 30 előre :méret/4
  tollatle
vége

```

```
eljárás halraj :méret :szint :db
  ha :szint>0 [tollatle
    ha maradék :szint 4<=1 [jobbra 180]
    hal :méret :szint+1 :szint+4
    ha maradék :szint 4<=1 [ jobbra 180]
    ismétlés :db [tollatfel előre :méret*1,5 tollatle
      halraj :méret/2 :szint-1 :db+1
      tollatfel hátra :méret*1,5
      jobbra 360/:db tollatle]]
vége
```

## Számításokkal vezérelt rajzolás

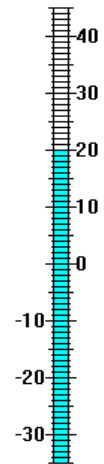
Időnként szükség lehet arra, hogy a rajz elkészítéséhez különböző számításokat is elvégezzünk. Néhány példán keresztül szeretnénk megvilágítani, hogy miről is van szó. Ha az a feladatunk, hogy rajzoljuk ki egy analóg óra számlapját a helyes mutató állásokkal akkor, ha az éjfél óta eltelt időt adjuk meg percekben, akkor ki kell számolni, hogy hányadik órában járunk és az utolsó egész órától hány perc telt el. Hasonlóan, ha ki akarjuk rajzolni egy decimális szám bináris megfelelőjét, először el kell végezni a számrendszerek közötti átváltást. Nézzük meg a néhány további példát is megvalósításukkal együtt!

### Kezdőpont transzformációja

#### Hőmérő

Egy hőmérő  $-35$  és  $+45$  fok közötti hőmérséklet mérésére alkalmas.

Készíts Logo eljárást (hőmérő: fok), amely kirajzolja a hőmérőt, fokenként beosztást rajzol rá, 5 fokenként hosszabb, a 0-ra végződő fokoknál pedig még hosszabb vonallal! A 0-ra végződő fokokat számmal is kiírja a hőmérő mellé, a negatívakat balra, a pozitívokat pedig jobbra. A hőmérő aljától a :fok magasságig a higanyszálat is belerajzolja.



```
eljárás hőmérő :fok
  tollatfel balra 90 előre 5 jobbra 90 hátra 35*4 tollatle
  téglalap 80*4 10 higany :fok
  vonal 3 ismétlés 4 [vonat 1] vonal 5
  ismétlés 7 [ismétlés 4 [vonat 1] vonal 3
    ismétlés 4 [vonat 1] vonal 5]
  ismétlés 4 [vonat 1] jobbra 90 hátra 3 előre 16 hátra 13
  balra 90 hátra 80*4 feliratoz -30 40
  tollatfel hátra 42*4 jobbra 90 előre 5 balra 90 tollatle
vége
```

A higanyszál aktuális magasságának kiszámításánál figyelembe kell venni, hogy negatív értékeket is mérünk, így egy eltolást kell alkalmaznunk. Mivel  $-35$  foknál lesz a szál magassága nulla, a transzformációban az aktuális fok értékhez hozzá kell adnunk  $35$ -öt.

```
eljárás higany :fok
  ismétlés 9 [jobbra 90 előre 1 balra 90 tollszín! 11
    előre (:fok+35)*4 hátra (:fok+35)*4 tollszín! 0]
  jobbra 90 hátra 9 balra 90
vége
```

```
eljárás téglalap :x :y
  ismétlés 2 [előre :x jobbra 90 előre :y jobbra 90]
vége
```

A feliratozásnál arra kell figyelni, hogy negatív értékeknél a hőmérő bal oldalán, pozitívaknál pedig a jobb oldalon kell elhelyezni a skála értékeket.

```
eljárás feliratoz :tol :ig
  előre 5*4+8
  hakülönben :tol<0 [tollatfel balra 90 előre 25 jobbra 90
    tollatle címke :tol tollatfel balra 90
    hátra 25 jobbra 90 tollatle]
  [tollatfel jobbra 90 előre 15 balra 90
    tollatle címke :tol tollatfel jobbra 90
    hátra 15 balra 90 tollatle]
  ha :tol<:ig [előre 5*4-8 feliratoz :tol+10 :ig]
vége
```

```

eljárás vonal :h
  jobbra 90 hátra :h előre 10+2*:h hátra 10+:h balra 90 előre 4
vége

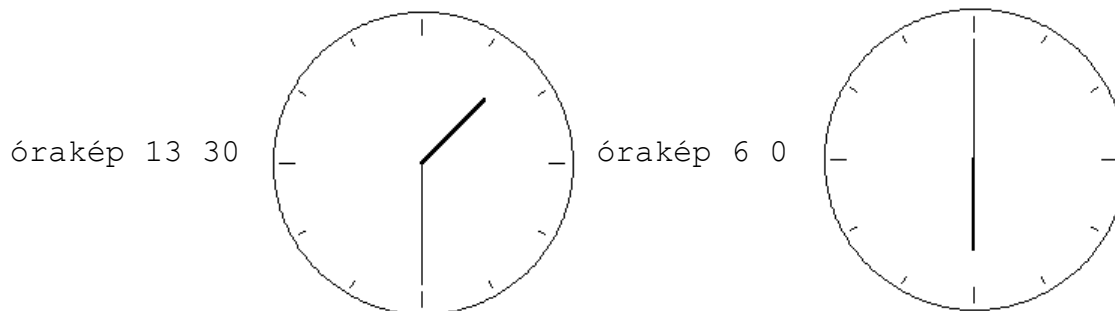
eljárás címke :hőm
  kiír :hőm
vége

```

## Elfordulás szögének kiszámítása

### Óra

Készíts órakép :o :p eljárást, amely az alábbi óralapot képes rajzolni, ahol az :o paraméter az óra értékét adja meg, a :p paraméter pedig a perceket.



A mutató körbefordulása közben 360 fokot tesz meg. Mivel egy órában 60 perc van, a nagymutató egy perc alatt 6 fokot fordul el. A kismutató 12 óra alatt fordul körbe, azaz fordul 360 fokot. Egy óra alatt a kismutató így 30 fokot, egy perc alatt 0,5 fokot fordul. Ezek alapján már kiszámítható egy adott időben az elfordulás szöge, ha meghatározzuk, hogy hány perc telt el éjfél óta (összesen hány fokot kellett elfordulnia).

```

eljárás órakép :o :p
  óralap 100 beosztás 100 mutató 80 1 6*:p
  mutató 60 2 (60*:o+:p)/2
vége

eljárás óralap :h
  tollatfel előre :h tollatle jobbra 90 körtrajzol 2*:h
  tollatfel balra 90 hátra :h tollatle
vége

eljárás beosztás :h
  ismétlés 4 [tollatfel előre :h-15 tollatle előre 10
             tollatfel hátra :h-5 tollatle jobbra 30
             tollatfel előre :h-10 tollatle előre 5
             tollatfel hátra :h-5 tollatle jobbra 30
             tollatfel előre :h-10 tollatle előre 5
             tollatfel hátra :h-5 tollatle jobbra 30]
vége

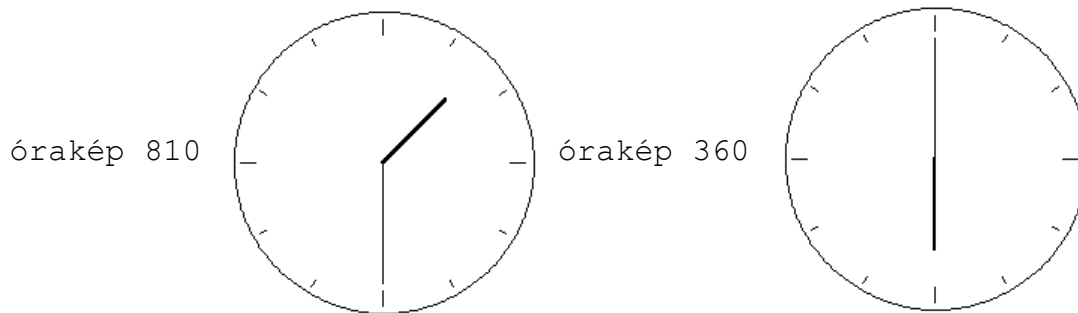
eljárás mutató :h :v :f
  tollvastagság! :v jobbra :f előre :h hátra :h balra :f
  tollvastagság! 1
vége

eljárás körtrajzol :h
  ismétlés 360 [előre 3,14159*:h/360 jobbra 1]
vége

```

## Óra, 2. variáció

Készíts órakép :perc eljárást, amely az alábbi óralapot képes rajzolni, ahol a :perc az éjfél óta eltelt perceket jelzi!



Felhasználva azt, hogy egy teljes kör bejárásához 360 fokot kell megtenni és azt, hogy a kismutató 12 óra alatt teszi meg ezt az utat, a nagymutató pedig 60 perc alatt, kiszámolhatóak az elfordulások, ahogy az előző feladatban is megtettük.

```

eljárás órákép :p
  óralap 100 beosztás 100 mutató 80 1 6*(maradék :p 60)
  mutató60 2 :p/2
vége

eljárás óralap :h
  tollatfel előre :h tollatle jobbra 90 körtrajzol 2*:h
  tollatfel balra 90 hátra :h tollatle
vége

eljárás beosztás :h
  ismétlés 4 [tollatfel előre :h-15 tollatle előre 10
             tollatfel hátra :h-5 tollatle jobbra 30
             tollatfel előre :h-10 tollatle előre 5
             tollatfel hátra :h-5 tollatle jobbra 30
             tollatfel előre :h-10 tollatle előre 5
             tollatfel hátra :h-5 tollatle jobbra 30]
vége

eljárás mutató :h :v :f
  tollvastagság! :v jobbra :f előre :h hátra :h balra :f
  tollvastagság! 1
vége

eljárás körtrajzol :h
  ismétlés 360 [előre 3,14159*:h/360 jobbra 1]
vége
  
```



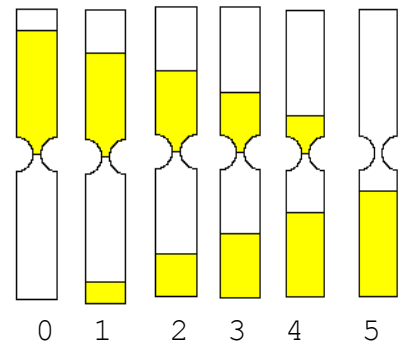
## Térfogatszámítás

### Homokóra

A homokórában kezdéskor az összes homok a felső tartályban van, majd minden perc elteltével egy egységgel fogy a mennyisége felülről és telik ugyanennyivel alul.

Készíts eljárást (homokóra :oldal :perc), amely egy 5 percig mérni tudó homokórát rajzol! Az :oldal legyen a homokóra szélessége, a többi méretet ehhez arányosan számold ki!

Figyeld meg az ábrán a percenkénti változást! Nem kell figyelned az üveg keskenyedő részére a számításnál, elég, ha azonos távolságokat veszel!



eljárás homokórákörvonal :a

előre 3\*:a jobbra 90 körb :a/5\*2 jobbra 90 előre 3\*:a jobbra 90

előre :a jobbra 90 előre 3\*:a jobbra 90 körb :a/5\*2 jobbra 90

előre 3\*:a jobbra 90 előre :a jobbra 90

vége

Balra forduló körív.

eljárás körb :r

ismétlés 180 [előre 2\*:r\*3,14159/360 balra 1]

vége

eljárás homokóra :oldal :perc

homokórákörvonal :oldal tollatle töltőszín! 14

hakülönben :perc=0 [felül :oldal 5]

[hakülönben maradék :perc 5=0

[alul :oldal 5]

[alul :oldal maradék :perc 5

felül :oldal 5-maradék :perc 5]]

tollatle

vége

A homokóra felső tartályában ennyi homok van még.

eljárás felül :a :perc

homokórákörvonal :a tollatle töltőszín! 14 tollatfel

előre 3\*:a+:a/5\*2 jobbra 90 előre :a/5\*2 tollatle előre :a/5

tollatfel hátra :a/5\*3 balra 90 tollatfel

előre :a/5\*2+:perc\*:a/2 tollatle jobbra 90 előre :a hátra :a

tollatfel jobbra 45 előre 5 tölt hátra 5 balra 45 balra 90

hátra :a/5\*2+:perc\*:a/2 hátra 3\*:a+:a/5\*2 tollatle

vége

A homokóra alsó tartályában ennyi homok van.

eljárás alul :a :perc

töltőszín! 14 előre :perc\*:a/2 jobbra 90 előre :a hátra :a

balra 90 hátra :a/2\*:perc tollatfel jobbra 45 előre 5 tölt

hátra 5 balra 45 tollatle tollatle

vége

### Kád

Egy négyzet alapú, adott magasságú kádba vizet töltünk. A kád fala három egység vastag. Ha a víz több, mint amennyi egyetlen kádba belefér, akkor újabb kádokba kerül a víz. A kádok az előzőtől jobbra három egység távolságra helyezkednek el.

Írd meg a kád :szél :mag :víz eljárást, amely megrajzolja a kád(ak)at! A kád szélességét és magasságát cm-ben, a vízmennyiséget literben adjuk meg!

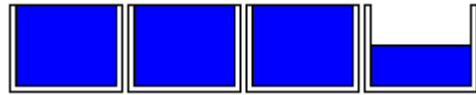
A kád térfogatát a  $V = \text{szél} * \text{szél} * \text{mag}$  képlettel számítjuk.



kád 100 80 800



kád 100 80 500



kád 50 40 350

```
eljárás üreskád :szél :mag
  jobbra 90 előre :szél balra 90
  előre :mag jobbra 90 előre 3 jobbra 90 előre :mag+3
  jobbra 90 előre :szél+6 jobbra 90
  előre :mag+3 jobbra 90 előre 3 jobbra 90 előre :mag
  balra 180
vége
```

A kád eljárás kiszámolja a térfogatot és ennek megfelelően feltölti a kádat. Ha a víz nem fér bele egy kádba, akkor újabb kádat rajzol.

```
eljárás kád :szél :mag :víz
  üreskád :szél :mag
  hakülönben :víz>:mag*:szél*:szél/1000
    [önt :szél :mag
      tollatfel jobbra 90 előre :szél+9 balra 90
      tollatle kád :szél :mag :víz-:mag*:szél*:szél/1000
      tollatfel jobbra 90 hátra :szél+9 balra 90 tollatle]
    [önt :szél :víz*1000/(:szél*:szél)]
vége
```

Az önt folyadékkal tölti fel a kádat.

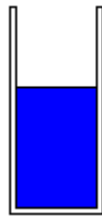
```
eljárás önt :szél :mag
  ismétlés 2 [előre :mag jobbra 90 előre :szél jobbra 90]
  töltőszín! 9
  jobbra 45 tollatfel előre 2 tölt hátra 2 balra 45 tollatle
vége
```

## Üdítő

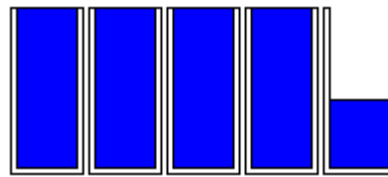
Üdítőt adagolunk henger alakú poharakba. Ha az üdítő mennyisége több mint amennyi egy pohárba befér, akkor újabb poharakat használunk, amelyek az előzőtől jobbra három egység távolságra helyezkednek el.

Írd meg az üdítő :r :mag :menny eljárást, amely megrajzolja a pohar(ak)at! Az :r a henger alapkörének sugara mm-ben, a :mag a pohár magassága mm-ben, a :menny pedig a kitöltendő üdítő mennyisége literben! A pohár fala 3 egység vastag. Az üdítőt színezéssel jelöljük!

A henger térfogatát a  $V = \text{sugár}^2 * \pi * \text{magasság}$  képlettel számítjuk.



üdítő 40 100 0,3



üdítő 30 80 1

A megoldás csak a térfogatszámításban különbözik az előzőtől, hiszen itt henger térfogatot kell számítani.

```

eljárás üdítő :r :mag :menny
  ürespohár :r :mag
  hakülönben :menny>:mag*:r*:r*3,14159/1000000
    [önt :r :mag tollatfel jobbra 90 előre :r+9 balra 90
      tollatle üdítő :r :mag :menny-:mag*:r*:r*3,14159/1000000
      tollatfel jobbra 90 hátra :r+9 balra 90 tollatle]
    [önt :r :menny*1000000/(:r*:r*3,14159)]
vége

eljárás ürespohár :szél :mag
  jobbra 90 előre :szél balra 90
  előre :mag jobbra 90 előre 3 jobbra 90 előre :mag+3
  jobbra 90 előre :szél+6 jobbra 90
  előre :mag+3 jobbra 90 előre 3 jobbra 90 előre :mag balra 180
vége

eljárás önt :szél :mag
  ismétlés 2 [előre :mag jobbra 90 előre :szél jobbra 90]
  töltőszín! 9 jobbra 45 tollatfel előre 1
  ha :mag>1 [tölt] hátra 1 balra 45 tollatle
vége
    
```

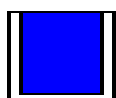
### Tejeskávé

Egy italautomata kávéból és tejből tejeskávét készít, a két alapanyagot 1:4 arányban keveri össze. Az automata a terméket henger alakú pohárba adagolja. A pohár fala 3 egység vastagságú. Ha valamelyik hozzávaló elfogyott, akkor az automata nem készít több adagot (ehhez poharat sem használ).

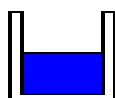
Készítsd el a tejeskávé: sugár: magasság: kávé: tej eljárást, ahol a kávé és a tej mennyiségét dl-ben, a henger alapkörének sugarát pedig mm-ben adjuk meg! A tejeskávét színezéssel jelöljük! Ha több pohárra van szükségünk, akkor a poharak 3 egység távolságra jobbra kerülnek az előzőtől.

A henger térfogatát a  $V = \text{sugár}^2 * \pi * \text{magasság}$  képlettel számítjuk.

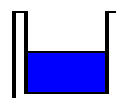
tejeskávé 20 40 :kávé :tej, ahol :kávé és :tej értékei az alábbiak:



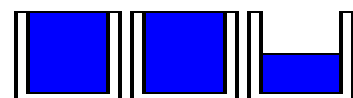
0,1 0,4



0,1 0,2



0,05 0,4



0,5 1

Addig rajzolunk újabb poharakat, amíg el nem fér a kávé mennyiség.

```

eljárás tejeskávé :r :mag :kávé :tej
  ha és (:kávé>0) (:tej>0) [ürespohár :r :mag]
  hakülönben és (:kávé>(térf :mag :r)/5)
    (:tej>(térf :mag :r)*4/5)
  [önt :r :mag tollatfel jobbra 90 előre :r+9
  balra 90 tollatle
  tejeskávé :r :mag :kávé-(térf :mag :r)/5
    :tej-(térf :mag :r)*4/5
  tollatfel jobbra 90 hátra :r+9 balra 90 tollatle]
  [hakülönben :kávé*4>:tej
  [önt :r :tej*125000/(:r*:r*3,14159)]
  [önt :r :kávé*500000/(:r*:r*3,14159)]]
vége
  
```

Adott magasságig önti a kávé.

```

eljárás önt :szél :mag
  ismétlés 2 [előre :mag jobbra 90 előre :szél jobbra 90]
  töltőszín! 9 jobbra 45 tollatfel előre 1
  ha :mag>1 [tölt]
  hátra 1 balra 45 tollatle
vége
  
```

```

eljárás ürespohár :szél :mag
  jobbra 90 előre :szél balra 90
  előre :mag jobbra 90 előre 3 jobbra 90 előre :mag+3
  jobbra 90 előre :szél+6 jobbra 90
  előre :mag+3 jobbra 90 előre 3 jobbra 90 előre :mag balra 180
vége
  
```

Kiszámolja a térfogatot.

```

eljárás térf :mag :r
  eredmény :mag*:r*:r*3,14159/100000
vége
  
```

## Számrendszerek közötti átváltás

### Bináris szám kirajzolása

Készíts Logo programot (számíró :szám), amely egy 0 és 255 közötti számot bináris számként, pálcikaként rajzolva tesz ki a képernyőre! Az egyes számjegyek 20 egység magasságúak, a 0 pedig 10 egység szélességű, s közöttük 5 egység távolság van. Ha a szám nem 0, akkor a bináris felírásban a bevezető 0-k ne jelenjenek meg.

```

0: □ 1: |
2: |□ 10: |□|□
129: |□□□□□□|
  
```

A decimális számrendszerből a bináris számrendszerre való átváltást úgy végezzük el, hogy kettővel osztjuk a decimális számot. A maradék képi a következő legkisebb helyiértékű bináris számjegyet (jobbról balra haladva), az eredménnyel pedig újra és újra elvégezzük az osztást, amíg nullát nem kapunk.

```

eljárás számíró :n
  hakülönben mod :n 2 = 0 [nulla][egy]
  ha :n>1 [tollatfel balra 90
    hakülönben mod :n 2 = 0 [előre 15][előre 5]
    jobbra 90 tollatle számíró egészhányados :n 2
    tollatfel balra 90
    hakülönben mod :n 2 = 0 [hátra 15][hátra 5]
    jobbra 90 tollatle]
vége

eljárás nulla
  ismétlés 2 [előre 20 balra 90 előre 10 balra 90]
vége

eljárás egy
  előre 20 hátra 20
vége
  
```

## Vegyes alapú számrendszerek közötti átváltás

### Címletező

A pénzkifizetéseknel érdemes a lehető legkevesebb darabszámú pénzzel kifizetni a kívánt összeget. A pénzeket (100, 200, 500, 1000, 2000, 5000, 10000, 20000 Ft) különböző színű téglalapokkal jelöljük (a színek a felsorolásban szereplő sorszámuk, azaz pl. a 100-as az 1-es kódú szín, ...), melyekre az összeget is felírjuk. *Vegyük észre, hogy itt egy vegyes alapú számrendszerről van szó, ahol a számrendszer alapszámait a pénzcímletek jelentik!*

Írd meg a címlet :ft eljárást, amely megrajzolja az :ft összeg kifizetéséhez szükséges pénzeket! Az azonos címleteket egymás mellé rajzold!

```

      500
      [500]
címlet 500

      1000 200 200
      [1000][200][200]
címlet 1400

20000 10000 5000 2000 1000 500 200 100
[20000][10000][5000][2000][1000][500][200][100]
címlet 38800
  
```

A nyolc pénzféle egyetlen eljárásban is elkészíthető.

```

eljárás pénz :szín :ft
  töltőszín! :szín
  ismétlés 4 [előre 20 jobbra 90 előre 50 jobbra 90]
  jobbra 45 tollatfel előre 10 tölt hátra 10 balra 45
  előre 17 jobbra 90 előre 3 balra 90 tollatle címke :ft
  tollatfel jobbra 90 hátra 3 balra 90 hátra 17 tollatle
vége
  
```

A címletezésnél mindig a lehető legnagyobb címletű pénzekből kell összeállítani az összeget! A megoldás menete is ezt a gondolatot követi, nagyság szerint csökkenő sorrendben megpróbálja az adott címletet felhasználni a kifizetéshez, majd a maradék pénzzel ugyanezt a folyamatot végigcsinálni, amíg a címletezendő pénz el nem fogy.

```

eljárás címlet :ft
  hakülönben :ft>=20000 [péNZ 8 20000 lép címlet :ft-20000]
  [hakülönben :ft>=10000 [péNZ 7 10000 lép címlet :ft-10000]
  [hakülönben :ft>=5000 [péNZ 6 5000 lép címlet :ft-5000]
  [hakülönben :ft>=2000 [péNZ 5 2000 lép címlet :ft-2000]
  [hakülönben :ft>=1000 [péNZ 4 1000 lép címlet :ft-1000]
  [hakülönben :ft>=500 [péNZ 3 500 lép címlet :ft-500]
  [hakülönben :ft>=200 [péNZ 2 200 lép címlet :ft-200]
  [ha :ft>=100 [péNZ 1 100 lép címlet :ft-100]]]]]]]]
  ha :ft>=100 [tollatfel jobbra 90 hátra 60 balra 90 tollatle]
vége

```

```

eljárás lép
  tollatfel jobbra 90 előre 60 balra 90 tollatle
vége

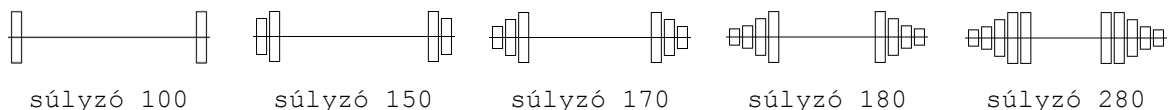
```

## Súlyzó

A súlyemelőknél a súlytárcsák 50, 25, 10 és 5 kg-osak (a rúd és a leszorítók súlyát most nem, vesszük figyelembe) lehetnek. A két oldalra mindig ugyanolyan súlynak kell kerülni, belülről kifelé csökkenő sorban. Adott súlyhoz a lehető legkevesebb súlytárcsát kell a súlyzóra tenni. *Vegyük észre, hogy itt is egy vegyes alapú számrendszert használunk, ahol az elérhető súlytárcsák adják meg az alapszámokat!*

Írd meg a súlyzó : kg eljárást, amely megrajzolja a megfelelő súlyokat a rúddal együtt! A súlyok 10 egység magasságú hengerek, az oldalukra állítva, közöttük 3 egység távolsággal. A súlyzó rúdja 200 egység hosszú, a kívánt súlyok biztosan rátehetőek.

A henger súlyát a  $SÚLY = sugár^2 * 3,14159 * magasság / 400$  képlettel számítjuk.



Mindkét irányban egyenletesen elosztja a súlyt.

```

eljárás súlyzó :súly
  balra 90 előre 200/2-13*darab :súly/2 súlyok :súly/2
  jobbra 180 előre 200-13*darab :súly/2 súlyok :súly/2
  hátra 200/2 balra 90
vége

```

A súlyok kirajzolását végzi.

```

eljárás súlyok :súly
  hakülönben :súly>=50 [rajzol 50 súlyok :súly-50]
  [hakülönben :súly>=25 [rajzol 25 súlyok :súly-25]
  [hakülönben :súly>=10 [rajzol 10 súlyok :súly-10]
  [ha :súly>=5 [rajzol 5]]]]
vége

```

Ennyi darab súlyt kell rátenni a rúdra.

```

eljárás darab :súly
  ha :súly=0 [eredmény 0]
  ha :súly>=50 [eredmény 1+darab :súly-50]
  ha :súly>=25 [eredmény 1+darab :súly-25]
  ha :súly>=10 [eredmény 1+darab :súly-10]
  eredmény 1
vége

```

Egy konkrét súly kirajzolása.

```
eljárás rajzol :súly
  jobbra 90
  ismétlés 2 [előre 20*gyök :súly/10/3,14159 balra 90 előre 10
              balra 90 előre 20*gyök :súly/10/3,14159]
  balra 90 előre 13
vége
```

### Római számok

Készíts Logo eljárást (római :szám), amely egy 1 és 10 közötti számot római számokkal ír ki a képernyőre! A megoldásban nem használhatod a betűket kiíró Logo utasításokat, így a betűket a teknőccel kell megrajzolni!

A római számok: 1- I, 2 – II, 3 – III, 4 – IV, 5 – V, 6 – VI, 7 – VII, 8 – VIII, 9 – IX, 10 – X.



A háromféle számjegy így nézzen ki:

Tudjuk, hogy egy és három között annyi vonalat kell rajzolni, amennyi a szám értéke. Négy esetében egy vonal és egy v alak, öt és öttől kilencig a v alak és annyi vonal, amennyivel nagyobb a szám ötnél, kilencnél egy vonal és egy x, tíznél pedig egy x az eredmény.

```
eljárás római :szám
  hakülönben :szám<4 [ismétlés :szám [i]]
    [hakülönben :szám=4 [i v]
      [hakülönben :szám<9 [v ismétlés :szám-5 [i]]
        [hakülönben :szám=9 [i xx][xx]]]]]
vége
```

```
eljárás i
  balra 90 előre 3 hátra 6 előre 3 jobbra 90 előre 50
  balra 90 előre 3 hátra 6 előre 3 jobbra 90 hátra 50
  tollatfel jobbra 90 előre 12 balra 90 tollatle
vége
```

```
eljárás v
  tollatfel előre 50 tollatle
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  balra 15 hátra 52 jobbra 30 előre 52 balra 15
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  tollatfel hátra 50 jobbra 90 előre 12 balra 90 tollatle
vége
```

```
eljárás xx
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  jobbra 15 előre 52 balra 15
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  jobbra 15 hátra 26 balra 30 előre 26 jobbra 15
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  balra 15 hátra 52 jobbra 15
  balra 90 előre 3 hátra 6 előre 3 jobbra 90
  tollatfel jobbra 90 előre 12 balra 90 tollatle
vége
```

## A környezet érzékelésével vezérelt rajzolás

A robotoknak különböző érzékelőik lehetnek. Például fény, mozgás, hő vagy hangérzékelő. Ha egy robotnak azt a feladatot akarjuk adni, hogy kövessen egy földre húzott vonalat, akkor ehhez érzékelnie kell a talaj színét, amiből megállapíthatja, hogy vonalon van vagy sem. Mozgásérzékelő esetén robotunkat arra kérhetjük, hogy végezzen el egy feladatot (mondjuk, nyissa ki az ajtót vagy készítsen egy fényképet és küldje le nekünk).

Logo teknőcünket is egy robotnak tekinthetjük, amelyet különböző feladatokra kérünk fel. Alap esetben a teknőcünk a környezetéből a saját helyét, irányát és a pont színét tudja érzékelni – ezt fogjuk kihasználni. Ilyenkor „tapogatózunk”, lekérjük az aktuális környezeti változót, és ha megfelelő, csak akkor folytatjuk tovább az utunkat.

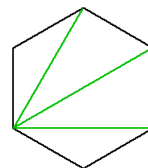
### Rajzolás vonalig

A következő feladatokban a rajzlap adott pontjának színét kérdezzük le és ez alapján haladunk tovább.

#### Átlós sokszög

Rajzolj szabályos `:db` oldalszámú, `:h` oldalhosszú fekete színű sokszöget átlós `:db` `:h`! Rajzold meg az egyik csúcsából kiinduló összes átlót, zöld színnel! (Segítség: a PONTSZÍN függvény megadja a teknőc aktuális pontjának színét.)

átlós 6 100



tollszín!

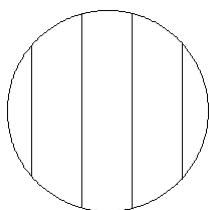
```
eljárás átlós :db :h
"fekete
ismétlés :db [előre :h jobbra 360/:db]
ismétlés :db-3 [jobbra (180-360/:db)/(:db-2) tollatfel
tollszín! "zöld előre 2 vonal hátra 2]
ismétlés :db-3 [balra (180-360/:db)/(:db-2)]
vége
```

Az átlók megrajzolásánál nem tudjuk azok hosszát, így addig rajzolunk, amíg a pontszín fekete nem lesz, vagyis el nem érjük a szemközti csúcsot.

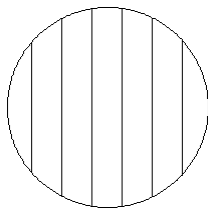
```
eljárás vonal
előre 1 hakülönben pontszín<>"fekete [vonaltollatle]
hátra 1
vége
```

#### Csíkos kör

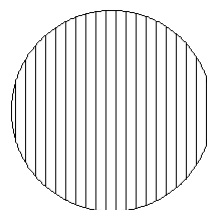
Rajzolj csíkos kört (csíkos `:r` `:sz`), ami egy `:r` sugarú kör, a középpontjára szimmetrikusan elhelyezett, egymástól `:sz` távolságra levő függőleges csíkokkal!



csíkos 100 50



csíkos 100 30



csíkos 100 10



Megrajzoljuk a körvonalat, majd becsíkozunk.

```
eljárás csíkos :r :sz
  kör 2*:r
  csíkok :r :sz 1
  csíkok :r :sz -1
  tollatle
vége
```

Függőleges csíkokat rajzolunk, a középpontból először felfelé, majd lefelé indítunk vonal rajzolást a körvonalig, majd átlépünk a következő csík helyére.

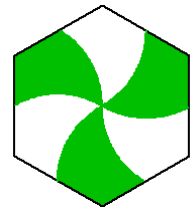
```
eljárás csíkok :r :sz :irány
  ha :r >0 [tollatfel vonal jobbra 180 tollatfel előre 1 vonal
            jobbra 180 tollatfel jobbra 90 előre :sz*:irány balra 90
            csíkok :r-:sz :sz :irány tollatfel
            jobbra 90*:irány hátra :sz balra 90*:irány tollatle]
vége
```

Nem tudjuk előre, hogy milyen hosszú vonalat kell rajzolni, így azt figyeljük, hogy mikor érjük el a körvonalat, ami feketével lett megrajzolva.

```
eljárás vonal
  ha pontszín<>"fekete [előre 1 vonal tollatle hátra 1 tollatfel]
vége
```

## Hatszög

Készíts Logo programot (ábra :h), amely az ábrán látható :h oldalhosszúságú hatszöget rajzolja! A hatszögben levő körívek egy  $h \cdot \sqrt{3}$  sugarú kör ívei. A körívek vonalai és a páronként köztük levő terület legyen zöld!



Segítség: a pontszín függvény megadja, hogy a teknőc éppen milyen színű ponton áll, a rajzlapszín függvény pedig azt, hogy milyen színű rajzlapra rajzol.

```
eljárás ábra :h
  tollszín! 0 tollvastagság! 2
  hatszög :h tollatfel jobbra 60 előre :h balra 60 tollatle
  ismétlés 6 [ív :h*gyök 3 jobbra 60]
  tollszín! 0 tollvastagság! 2 töltőszín! "zöld jobbra 30
  ismétlés 3 [tollatfel előre :h/2 tölt hátra :h/2 tollatle
              jobbra 120]
vége
```

Ívek esetében a kerekítési pontatlanságok miatt az eredeti pozícióba való visszatéréshez a következő trükköt lehet alkalmazni: paraméterként adjuk át a kiindulási pozíciót és irányt! Írjuk meg a módosított ábra eljárást! (Meghívása: ábra2 100 poz irány).

```
eljárás ábra2 :h :poz :irány
  tollszín! 0 tollvastagság! 2
  hatszög :h tollatfel jobbra 60 előre :h balra 60 tollatle
  ismétlés 6 [ív :h*gyök 3 jobbra 60]
  tollszín! 0 tollvastagság! 2 töltőszín! "zöld jobbra 30
  ismétlés 3 [tollatfel előre :h/2 tölt hátra :h/2 tollatle jobbra 120]
  tollatfel poz! :poz irány! :irány
vége

eljárás hatszög :h
  ismétlés 6 [előre :h jobbra 60]
vége
```

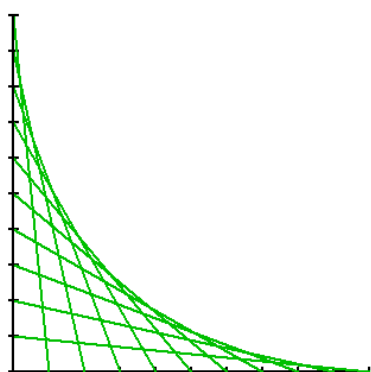
Először „bejárjuk” az utat felemelt tollal addig, amíg meg nem találjuk a hatszög oldalát, majd a rekurzió segítségével visszahátrálunk a hatszög közepére.

```

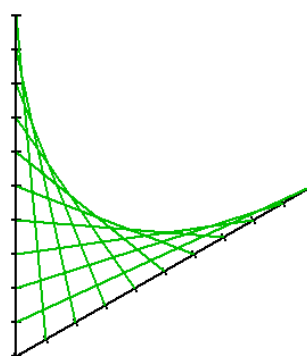
eljárás ív :r
  tollatfel előre :r*3,14159/180 jobbra 1
  ha pontszín=háttérszín [ív :r]
  tollatle balra 1 hátra :r*3,14159/180
vége
  
```

## Ívek

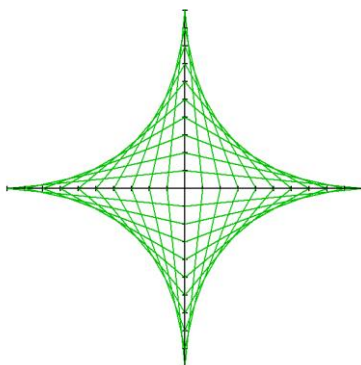
Az alábbi három ábrához (ábra1:n :h :szög, ábra2 :n :h, ábra3 :n :h :szög) szükség van koordináta-tengelyekre, melyeken :n darab :h hosszúságú egység szerepel. A beosztásokat az ábrának megfelelően kötjük össze. A második és a harmadik ábra az elsőből 4 darabot tartalmaz. A második ábránál a két tengely 90 fokos szöget zár be, az első és harmadik ábránál pedig megadható a :szög szög. Készítsd el a három ábrarajzoló eljárást!



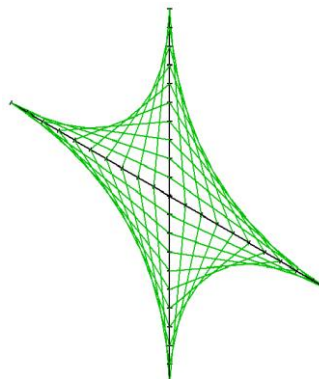
ábra1 10 40 90



ábra1 10 40 60



ábra2 10 20



ábra3 10 20 120

Vegyük észre, hogy az egyik tengelyen lévő pontból kell a másik tengely minden pontjába vonalat húzni! Ezt a műveletet kell elvégezni mindkét tengelyre. Ilyenkor csak a másik tengelyhez való elfordulás szöge ellentétes.

```

eljárás ábra1 :n :h :szög
  előre :n*:h
  tengely :n*:h :n :h :szög poz irány
  hátra :n*:h jobbra :szög előre :n*:h
  tengely :n*:h :n :h -1*:szög poz irány
  hátra :n*:h balra :szög
vége
  
```

A tengely eljárás tehát az egyik tengely végpontjába húz vonalakat a másik tengely minden pontjából. Azt, hogy milyen irányba kell elindulni az adott ponthoz, az irányyszög függvényével határozzuk

meg. Azt, hogy meddig kell előre lépünk, azt pedig az eddigi előre „tapogatózó” módszerrel találjuk meg.

```

eljárás tengely :teljeshossz :n :h :szög :poz :irány
  ha :n>0 [hátra :teljeshossz jobbra :szög előre :n * :h
    irány! irányszög :poz tollatfel tollszín! "zöld vonal
    irány! :irány poz! :poz tollszín! "fekete
    tengely :teljeshossz :n-1 :h :szög :poz :irány]

```

vége

```

eljárás vonal

```

```

  ha pontszín<>"fekete [előre 1 vonal tollatle hátra 1 tollatfel]

```

vége

Az ábra2 egyszerűen az ábra1 négyszer megforgatott változata 90 fokos szöggel.

```

eljárás ábra2 :n :h

```

```

  ismétlés 4 [ábra1 :n :h 90 jobbra 90]

```

vége

Az ábra3 pedig két-két :szög és 180-:szög paraméterrel meghívott ábra1 eljárással készíthető.

```

eljárás ábra3 :n :h :szög

```

```

  ismétlés 2 [ábra1 :n :h :szög jobbra :szög

```

```

    ábra1 :n :h 180-:szög jobbra 180-:szög]

```

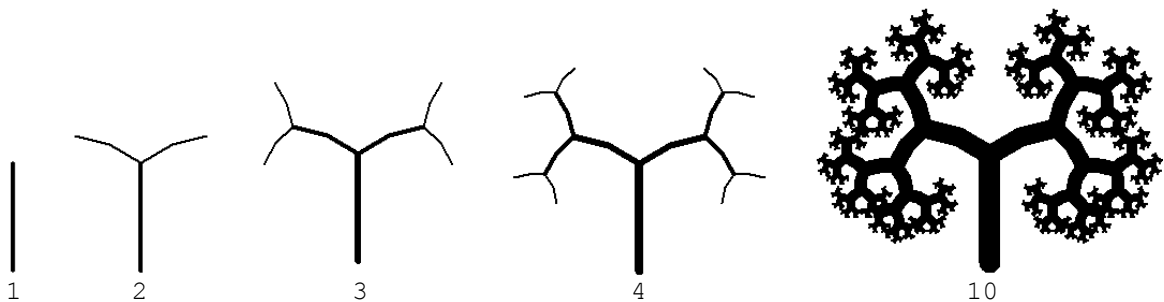
vége

## Irányfüggő rajzok

Az előző feladatokkal ellentétben most a rajzoló teknőc irányától függően haladunk tovább. Több ilyen feladattal is találkozhatunk a fákkal foglalkozó fejezetben. Egyet itt is megismétlünk.

## Irányfüggő fa

A fa függőlegestől balra hajló ágai közepükön 15 fokkal balra, a jobbra hajlók pedig 15 fokkal jobbra hajlanak!



```

eljárás fa24 :n :h

```

```

  tollvastagság! :n*2 előre :h/2

```

```

  hakülönben vagy egyenlő? irány 0 egyenlő? irány 180

```

```

  [fa24a :n :h]

```

```

  [hakülönben irány<180 [jobbba 15 fa24a :n :h balra 15]

```

```

    [balra 15 fa24a :n :h jobbba 15]]

```

```

  tollvastagság! :n*2 hátra :h/2

```

vége

```

eljárás fa24a :n :h

```

```

  előre :h/2

```

```

  ha :n>1 [balra 60 fa24 :n-1 :h*2/3 jobbra 120 fa24 :n-1 :h*2/3 balra 60]

```

```

  hátra :h/2

```

vége

## Szöveggel, listákkal vezérelt rajzolás

A Logo nyelv maga egy automata elvű (rajzoló) illetve egy funkcionális elvű (szövegkezelő) programozási nyelvből áll össze. A rajzoló programozási lehetőségek megismerése után kézenfekvő a szöveggel vezérelt rajzolási feladatokon keresztül eljutni a funkcionális nyelvi lehetőségek megismeréséig. Az informatikán belül az egyik jelentős terület a különböző vezérlési feladatok megoldása mondjuk egy gyártósoron, de gondolhatunk például egy robotra a jövőből, amelyik már érti, amire szóban utasítjuk.

Eddigi feladatainkban a rajzaink paraméterszáma ismert volt már a fejlesztési idő során – tudtuk előre, hogy pontosan milyen alakú ábrát kívánunk rajzolni, milyen színnel. Amennyiben ezeket az adatokat nem tudjuk előre, akkor valamilyen összetett adatszerkezetre, sorozatra van szükségünk a rajz elkészítéséhez. Az Imagine Logo-ban a sorozatokat listaként lehet ábrázolni, amelyek első eleméhez illetve első nélküli elemeihez lehet hozzáférni. Egy szót, mondatot, szöveget magát is listaként lehet felfogni – karakterek, szavak vagy mondatok listájaként.

A szövegben, szövegekben különböző parancsokat írhatunk le, amelyeket akár futás közben, interaktívan is kiadhatunk. Így készíthetünk például egyszerű rajzoló programot vagy akár rovásíró alkalmazást is. A szövegekben megadhatjuk egy sor vagy mozaik kirajzolható elemeit, ilyenkor természetesen nem kell ezeknek valamilyen szabályszerűségnek megfelelően következni egymás után. Lássunk néhány példát szöveggel irányított rajzolásra.

### Interaktív vezérlés karakterekkel

#### Tanulás

Írj Logo eljárást `tanulás`, amellyel egy rajzot megtaníthatunk a teknőcnek, majd kirajzoltathatjuk vele újra, illetve kirajzoltathatjuk a rajz tükörképét is!

A teknőcöt a `JBEH` parancsokkal vezéreljük. **J** hatására jobbra, **B** hatására balra fordul 90 fokkal, **E** hatására előre lép egyet, **H** hatására pedig hátra. A beírt parancsokat rajzolás közben megjegyezzük mindaddig, amíg **V** betűt nem nyomunk. A **T** betűre töröljük a képernyőt! Ezután az **R** betű lenyomására rajzoljuk ki a megjegyzett rajzot, az **A** betű hatására az y-tengelyre vett tükörképét, a **B** betű hatására az x-tengelyre vett tükörképét, a **C** betűre pedig az origóra vett tükörképét! A következő **V** betűre a program fejeződjön bel!

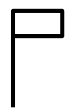
```
Tanulás EEEEEEEJEEEEBHHBEEEEJV
```

Vezérli a teknőst a billentyűzetről beolvasott karakterek alapján. Az `olvasjel` parancs egy karakter beolvasását végzi el.

```
eljárás tanulás
  rajzol feldolgoz olvasjel olvasjel
vége
```

A rajzoló parancsokra elvégzi a megfelelő utasításokat.

```
eljárás rajzol :szó :kar
  ha :kar="T [törölkép]
  ha :kar="R [kirajzol :szó]
  ha :kar="A [kirajzol tükröz :szó]
  ha :kar="B [kirajzol tükrözx :szó]
  ha :kar="C [kirajzol tükröz0 :szó]
  ha :kar">"V [rajzol :szó olvasjel]
vége
```



Egy alaputasítás sorozatnak megfelelően irányítja a teknőcöt.

```

eljárás kirajzol :szó
  ha nem üres? :szó [hakülönben első :szó="J [jobbra 90] []
                    hakülönben első :szó="B [balra 90] []
                    hakülönben első :szó="E [előre 10] []
                    hakülönben első :szó="H [hátra 10] []
                    kirajzol elsőnélküli :szó]
vége
  
```

Egy karakter által megadott alaputasítást hajt végre.

```

eljárás feldolgoz :kar
  ha :kar="V [eredmény []]
  ha :kar="J [jobbra 90]
  ha :kar="B [balra 90]
  ha :kar="E [előre 10]
  ha :kar="H [hátra 10]
  eredmény elsőnek :kar feldolgoz olvasjel
vége
  
```

Origóra tükrözés.

```

eljárás tükrözo :szó
  ha üres? :szó [eredmény :szó]
  ha első :szó="E [eredmény elsőnek "H tükrözo elsőnélküli :szó]
  ha első :szó="H [eredmény elsőnek "E tükrözo elsőnélküli :szó]
  eredmény elsőnek első :szó tükrözo elsőnélküli :szó
Vége
  
```

Y irányú tükrözés:

```

eljárás tükrözy :szó
  ha üres? :szó [eredmény :szó]
  ha első :szó = "B [eredmény elsőnek "J tükrözy elsőnélküli :szó]
  ha első :szó = "J [eredmény elsőnek "B tükrözy elsőnélküli :szó]
  eredmény elsőnek első :szó tükrözy elsőnélküli :szó
vége
  
```

X irányú tükrözés előállítható az origóra tükrözés és az x tengelyre tükrözések egymásutánjával.

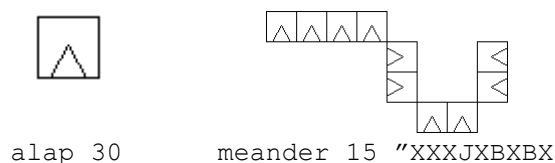
```

eljárás tükrözx :szó
  eredmény tükrözy tükrözo :szó
vége
  
```

## Vezérlés karaktorsorozattal

### Meander, fordulások vezérlése

Meandernek nevezik az olyan sormintákat, amelyek valamilyen szabályszerűség szerint kanyarognak. Készíts Logo eljárást (alap :h), amely egy, az alábbi ábrának megfelelő alapelemet rajzol! A meander kanyargását paraméterekkel szeretnénk vezérelni. Készíts Logo eljárást (meander :h :sz), amely az alapelemet az :sz szöveg karakterei szerinti sorrendben ismétli. A meander először jobbra indul. Ha az **X** betű következik, akkor a haladási irányt megtartja; a **J** betű hatására az irány jobbra változik 90 fokkal, a **B** hatására pedig balra 90 fokkal.



Karakterenként rajzolja ki a meandert!

```
eljárás meander :h :sz
  hakülönben nem üres? :sz [alap :h mozdít :h első :sz
                             meander :h elsőnélküli :sz]
                             [alap :h]
vége
```

Az elemi alakzat kirajzolása:

```
eljárás alap :h
  ismétlés 4 [előre :h jobbra 90]
  jobbra 90 előre :h/5 balra 60
  ismétlés 2 [előre 3*:h/5 jobbra 120]
  hátra :h/5 előre :h jobbra 90
vége
```

A beérkező parancs kódja alapján kanyarodunk:

```
eljárás mozdít :b
  ha :b="X [jobbra 90 előre :h balra 90]
  ha :b="J [jobbra 90 előre :h]
  ha :b="B [jobbra 90 előre :h balra 90 előre :h balra 90 hátra :h]
vége
```

### Ovilogo, a fordulások és a lépések nagyságának vezérlése

Az előző interaktív rajzoló programhoz hasonló feladattal állunk szemben. A korábbi karakterről karakterre megadott parancsokkal szemben most egyszerre, egy karakterekből álló listával/szóval adjuk meg a parancssorozatot! Az első osztályosok számára készült Logo nyelv egyszerűbb az általunk használnál. Összesen 4 utasítást, valamint egyjegyű egész számokat ismer:

**E:** előre lép 10\*X egységet **H:** hátra lép 10\*X egységet

**B:** balra fordul 90 fokot **J:** jobbra fordul 90 fokot

**0..9:** a lépés egységét (X-et) az adott számjegyre változtatja

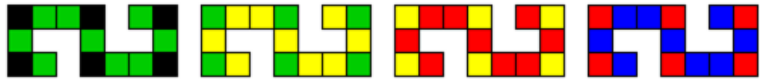
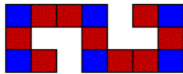
Készíts Logo eljárást OVI "szöveg, amely a szöveggént megadott „Ovilogo” programot vérehajtja, eredményét a képernyőre rajzolja!

```
ovi "EEJEEEEJ3E5HB1E
eljárás ovi :szó
  törölkép feldolgoz :szó 1
vége
eljárás feldolgoz :szó :x
  ha nem üres? :szó
    [hakülönben szám? első :szó
     [feldolgoz elsőnélküli :szó első :szó]
     [csinál első :szó :x feldolgoz elsőnélküli :szó :x]]
vége
eljárás csinál :betű :x
  hakülönben :betű="E [előre :x*10]
  [hakülönben :betű="H [hátra :x*10]
  [hakülönben :betű="B [balra 90]
  [ha :betű="J [jobbra 90]]]]
vége
```



## Görögös sorminta, színezés vezérlése

Készítsd el az alábbi sormintát lerajzoló görögösor :oldal :leírás nevű eljárást, amelynek váltakozó színeit egy szöveges paraméterrel (:leírás) adhatod meg, az alapelem hosszát pedig az :oldal paraméterrel. A sor egy eleme négyzetekből álló minta, amely két különböző színnel van kiszínezve görögös :oldal :szín1 :szín2. Mindig a sarokszínek azonosak egy sorelemben. Figyeld meg, hogy két egymást követő elemben a második elem sarokszíne, az első elem kitöltő színe volt és az új kitöltőszínt a leíró paraméterből olvassuk ki. A :leírás-ban a színek rendre p=piros, f=fekete, z=zöld, s=sárga, k=kék.



görögös 20 12 9

görögösor 20 "fzspk

```
eljárás görögös :oldal :szín1 :szín2
  négyzet :oldal :szín2 jobbra 90 előre :oldal balra 90
  négyzet :oldal :szín1 jobbra 90 hátra :oldal balra 90
  előre :oldal négyzet :oldal :szín1 előre :oldal
  négyzet :oldal :szín2
  ismétlés 2 [jobbra 90 előre :oldal balra 90 négyzet :oldal :szín1]
  jobbra 90 előre :oldal balra 90 négyzet :oldal :szín2
  hátra :oldal négyzet :oldal :szín1 hátra :oldal
  négyzet :oldal :szín2
  ismétlés 2 [jobbra 90 előre :oldal balra 90 négyzet :oldal :szín1]
  jobbra 90 előre :oldal balra 90 négyzet :oldal :szín2
  előre :oldal négyzet :oldal :szín1 előre :oldal
  négyzet :oldal :szín2
  balra 90 előre :oldal jobbra 90 négyzet :oldal :szín1 tollatfel
  hátra 2*:oldal jobbra 90 hátra 5*:oldal balra 90 tollatle
vége
```

```
eljárás négyzet :oldal :szín
  ismétlés 4 [előre :oldal jobbra 90]
  tollatfel töltőszín! :szín
  jobbra 45 előre 5 tölt hátra 5 balra 45 tollatle
vége
```

A sorminta színezésénél a leírásban szereplő betűk helyett a nekik megfelelő színnek kell szerepelnie, erre szolgál a szín függvény.

```
eljárás görögösor :oldal :leírás
  ha nem üres? elsőnélküli :leírás
  [görögös :oldal szín első :leírás szín első elsőnélküli :leírás
  tollatfel jobbra 90 előre 8*:oldal balra 90 tollatle
  görögösor :oldal elsőnélküli :leírás
  tollatfel jobbra 90 hátra 8*:oldal balra 90 tollatle]
vége
```

```
eljárás szín :szín
  ha :szín="f" [eredmény "fekete]
  ha :szín="z" [eredmény "zöld]
  ha :szín="k" [eredmény "kék]
  ha :szín="s" [eredmény "sárga]
  ha :szín="p" [eredmény "piros]
vége
```

### Képrajzoló, pozíció vezérlése

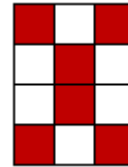
A kép `:sor :oszlop :méret :lista` eljárás egy `:sor* :oszlop`-os négyzetrács bizonyos elemeit színezi. A listában `k` jelenti a pirosra kiszínezendő mezőket, `f` a fehéren maradókat. A lista első eleme a bal alsó sarokban lévő négyzetet jelenti.



kép 3 2 15  
[k f f f k f]



kép 2 3 30  
[k f k f k f]



kép 4 3 20  
[k f k f k f f k f k f k]

```

eljárás kép :sor :oszlop :méret :lista
ismétlés :sor [ismétlés :oszlop [ismétlés 4 [előre :méret jobbra 90]
                                     jobbra 90 előre :méret balra 90]
                                     jobbra 90 hátra :oszlop* :méret balra 90 előre :méret]
hátra :sor* :méret
tollatfel jobbra 45 előre :méret/2 balra 45
festi :sor :oszlop :méret :lista 1
hátra :sor* :méret jobbra 45 hátra :méret/2 balra 45 tollatle
vége
    
```

A lista alapján kifesti a négyzetrács elemeit. A festésnél balról jobbra, alulról felfelé haladunk, egy-egy cellányit lépve. Ha elértünk a sor végére, visszatérünk a sor elejére és fellépünk egy sort!

```

eljárás festi :s :o :méret :li :x
ha nem üres? :li
  [hakülönben első :li="k [tölt] []
  hakülönben :x<:o [jobbra 90 előre :méret balra 90
                    festi :s :o :méret elsőnélküli :li :x+1]
                    [jobbra 90 hátra (:o-1)* :méret balra 90 előre :méret
                    festi :s :o :méret elsőnélküli :li 1]]
vége
    
```

### Morze, a rajzolás alakjának vezérlése

A Morze ábécében az egyes betűket hosszú és rövid vonalakkal jelöljük. Az egyes karaktereknek megfelelő pontok és vonalak kirajzolását vezéreljük majd a parancsunkkal. A következő feladatban az alábbi betűket használjuk:

a: • - o: - - - p: • - - • r: • - • t: -

Készíts Logo eljárásokat (`abetu`, `obetu`, `pbetu`, `rbetu`, `tbetu`) a fenti 5 betű morzejelének kirajzolására, valamint egy `szórajzol :szó` eljárást, amely egy szó morzejeleit rajzolja úgy, hogy az egyes betűk közé két üres helyet tesz!

jobbra 90 szórajzol "por -> • - - • - - - • - •

jobbra 90 szórajzol "tar -> - • - • - •

Karakterenként dolgozzuk fel a szót – karakterenként rajzoljuk ki a morzejeleket.

```

eljárás szórajzol :szó
ha nem üres? :szó [betű első :szó szünet
                  szórajzol elsőnélküli :szó]
vége
    
```



```

eljárás betű :b
  ha :b="a [rövid hosszú]
  ha :b="o [hosszú hosszú hosszú]
  ha :b="p [rövid hosszú hosszú rövid]
  ha :b="r [rövid hosszú rövid]
  ha :b="t [hosszú]

```

vége

```

eljárás rövid
  előre 1 szünet
vége

```

```

eljárás hosszú
  előre 5 szünet
vége

```

```

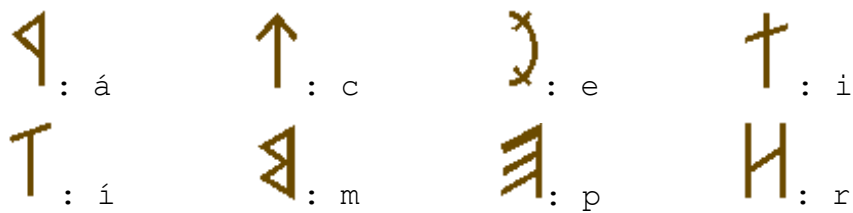
eljárás szünet
  tollatfel előre 5 tollatle
vége

```

## Rovásírás

A paraméterként megadott karakterek itt is a kirajzolandó ábra alakját vezérli. A rovásírás betűi egyenes vonalakból állnak.

Készíts Logo eljárást (rovás :szó), amely az alábbi betűket tartalmazó szavakat tud a képernyőre rajzolni! A betűk magassága azonos legyen! *Figyelj rá, hogy a rovásírásnál az írás iránya jobbról balra mutat!*



rovás "rím"      rovás "perc"      ↑H)≡

```

eljárás aabetu :h
  előre :h ismétlés 2 [balra 120 előre :h/2]
  balra 120 hátra :h/2
vége

```

```

eljárás cbetu :h
  előre :h balra 45 hátra :h/3 előre :h/3
  jobbra 90 hátra :h/3 előre :h/3 balra 45 hátra :h
vége

```

```

eljárás ibetu :h
  előre :h hátra :h/4 balra 105 előre :h/4 hátra :h/2 előre :h/4
  jobbra 105 hátra 3*:h/4
vége

```

```

eljárás iibetu :h
  előre 9*:h/10 balra 105 előre :h/4 hátra :h/2 előre :h/4
  jobbra 105 hátra 9*:h/10
vége

```

```

eljárás mbetu :h
  előre :h ismétlés 2 [balra 120 előre :h/2]
  jobbra 120 ismétlés 2 [előre :h/2 balra 120]
vége

```

```
eljárás pbetu :h
  előre :h balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3 balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3 balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3
vége

eljárás rbetu :h
  előre :h hátra 3*:h/5 jobbra 66 előre :h/2 balra 66
  előre 2*:h/5 hátra :h előre 3*:h/5 jobbra 66 hátra :h/2
  balra 66 hátra 2*:h/5
vége

eljárás ebetu :h
  jobbra 45 ismétlés 90 [előre :h*3,14159/270 balra 1]
  ismétlés 15 [jobbra 1 hátra :h*3,14159/270]
  balra 90 előre :h/6 hátra :h/3 előre :h/6 jobbra 90
  ismétlés 60 [jobbra 1 hátra :h*3,14159/270]
  balra 90 előre :h/6 hátra :h/3 előre :h/6 jobbra 90
  ismétlés 15 [jobbra 1 hátra :h*3,14159/270] balra 45
vége

eljárás rovas :szó
  ha nem üres? :szó [betű utolsó :szó 100 rovas utolsónélküli :szó]
vége

eljárás betű :b :h
  ha :b="á" [tollatfel jobbra 90 előre :h/2 balra 90 tollatle
    aabetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="c" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    cbetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="e" [tollatfel jobbra 90 előre :h/10 balra 90 tollatle
    ebetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="i" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    ibetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="í" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    iibetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="m" [tollatfel jobbra 90 előre :h/2 balra 90 tollatle
    mbetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="p" [tollatfel jobbra 90 előre 3*:h/5 balra 90 tollatle
    pbetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="r" [tollatfel jobbra 90 előre :h/10 balra 90 tollatle rbetu :h
    tollatfel jobbra 90 előre :h/2 balra 90 tollatle]
vége
```

## Titkosítás, alak és színek vezérlése

Achiram úgy készített titkosítást, hogy a betűket egy táblázatba helyezte és a betűnek megfelelő mező szegélyvonalát rajzolta le.

a	j	s		b	k	t		c	l	u
d	m	v		e	n	w		f	o	x
g	p	y		h	q	z		i	r	

Készíts Logo programot (achiram :szó), amely titkosítást készít egy ilyen kulcstábla alapján! Az azonos részben lévő betűket (pl. a j s) úgy különböztetjük meg, hogy az első betű fekete, a második piros, a harmadik pedig kékszínű legyen! (Így például a cica szó csupa fekete jellel rajzolható.) A program a beírt szöveget rajzolja meg a szabályok szerint!

```

achiram "cica
          L L L L
          U U U U
archiram "kata
          U U U U
    
```

```

eljárás achiram :szó
  ha nem üres? :szó [rajzol ascii első :szó
                    achiram elsőnélküli :szó]
vége
    
```

A betűk kódolva vannak. A kódok egymás után következő egész számok. Az a betű kódja 97! Ennek megfelelően a 9-cel való osztással lehet a betű helyét a táblázatban meghatározni.

```

eljárás rajzol :kód
  vonal maradék (:kód-97) 9 egészhányados (:kód-97) 9
vége
    
```

A mezőn belüli betűsorszám alapján választunk színt! A betűsorszámból az is kiszámolható, hogy a táblázat melyik mezőjébe kell elhelyezni, azaz most milyen keretet kell kirajzolni.

```

eljárás vonal :a :sz
  hakülönben :sz=0 [tollszín! 0]
                    [hakülönben :sz=1 [tollszín! 4][tollszín! 1]]
  tollvastagság! 5
  ha :a<6 [jobbra 90 előre 20 hátra 20 balra 90]
  ha :a>2 [tollatfel előre 20 tollatle jobbra 90 előre 20
           hátra 20 balra 90 tollatfel hátra 20 tollatle]
  ha (maradék :a 3)<>0 [előre 20 hátra 20]
  ha (maradék :a 3)<>2 [tollatfel jobbra 90 előre 20 tollatle
                       balra 90 előre 20 hátra 20 jobbra 90
                       tollatfel hátra 20 balra 90 tollatle]
  tollatfel jobbra 90 előre 30 balra 90 tollatle
vége
    
```

## Vezérlés számok listájával

### Minta, alak vezérlése

Egy tároló eszközön mélyebb, illetve kevésbé mélynyomással tárolják az információt.

Készíts Logo programot ábra :lista, amely a paraméterként kapott 0-kból, illetve 1-esekből álló sorozat esetén kirajzolja a tárolás ábráját!

Az ábrán a 0-nak a mélyebb, az 1-nek pedig a magasabb helyek felelnek meg: Az ábra a [0 1 0 1 0 1 0 0 0 1 1 1 0 1 0] sorozat alapján készült:



```

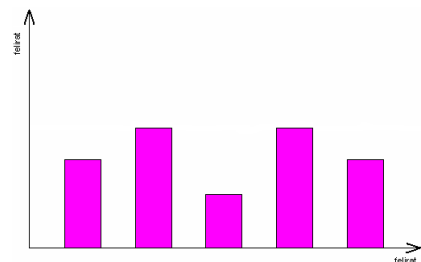
eljárás ábra :s
  jobbra 90 alul :s
  vége

eljárás alul :s
  ha nem üres? :s [hakülönben első :s=0 [előre 10 alul elsónélküli :s]
                  [balra 90 előre 10 jobbra 90
                  előre 10 felül elsónélküli :s]]
  vége

eljárás felül :s
  ha nem üres? :s [hakülönben első :s=1 [előre 10 felül elsónélküli :s]
                  [jobbra 90 előre 10 balra 90
                  előre 10 alul elsónélküli :s]]
  vége
  
```

### Időjárás, méret vezérlése

Készíts Logo programot (időjárás :lista), amely a listában megadott hőmérsékleti értékeket oszlopdiagramon ábrázolja lila színű téglalapokkal! A tengelyeket lásd el megfelelő felirattal! Az oszlopok szélessége és távolsága 20 egység legyen! A diagramra 10 oszlop férjen rá!



```

időjárás [50 70 30 70 50]

eljárás időjárás :s
  tengely 100 15 jobbra 90 tengely 20+40*elemszám :s (-5)
  oszlopok :s balra 90
  vége

eljárás tengely :h :i
  előre :h balra 30 hátra 10 előre 10 jobbra 60 hátra 10
  előre 10 balra 30 hátra 30
  tollatfel balra 90 előre :i betű! [Times New Roman][8 400 0]
  címke "felirat hátra :i jobbra 90 tollatle hátra :h-30
  vége

eljárás oszlopok :s
  előre 20 külső 20 első :s előre 20
  ha 1<elemszám :s [oszlopok elsónélküli :s]
  hátra 40
  vége

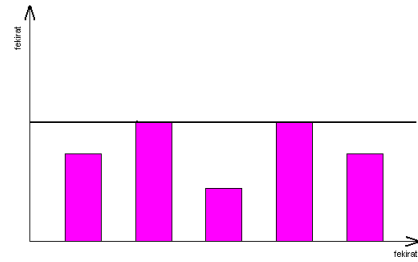
eljárás külső :x :y
  ismétlés 2 [előre :x balra 90 előre :y balra 90]
  tollatfel balra 45 előre 10 töltőszín! 13 tölt hátra 10
  jobbra 45 tollatle
  vége
  
```

## Időjárás, 2. verzió

Készíts Logo programot időjárás :lista, amely a listában megadott hőmérsékleti értékeket oszlopdiagramon ábrázolja lila színű téglalapokkal! A tengelyeket lásd el megfelelő felirattal! Az oszlopok szélessége és távolsága 20 egység legyen! A diagramra 10 oszlop férjen rá! A legmagasabb értékhez húzz egy vízszintes szintvonalat is!

időjárás [50 70 30 70 50]

A szintvonal meghatározásához ki kell számolni a maximumot!



```

eljárás időjárás :s
  tengely 100 15 jobbra 90 tengely 20+40*elemszám :s (-5)
  oszlopok :s szintvonal maximum :s elemszám :s
  vége

eljárás maximum :s
  hakülönben 1=elemszám :s [eredmény első :s]
  [eredmény nagyobb első :s maximum elsőnélküli :s]
  vége

eljárás nagyobb :a :b
  hakülönben :a>:b [eredmény :a] [eredmény :b]
  vége

eljárás szintvonal :x :y
  előre :x jobbra 90 előre 20+40*:y
  hátra 20+40*:y balra 90 hátra :x
  vége

eljárás tengely :h :i
  előre :h balra 30 hátra 10 előre 10 jobbra 60 hátra 10
  előre 10 balra 30 hátra 30
  tollatfel balra 90 előre :i betű! [Times New Roman][8 400 0]
  címke "felirat hátra :i jobbra 90 tollatle hátra :h-30
  vége

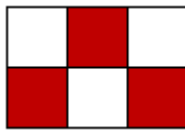
eljárás oszlopok :s
  előre 20 külső 20 első :s előre 20
  ha 1<elemszám :s [oszlopok elsőnélküli :s]
  hátra 40
  vége
  
```

## Képrajzoló, pozíció vezérlése

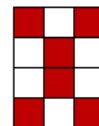
A kép :sor :oszlop :méret :lista eljárás egy :sor\*:oszlop-os négyzetrács bizonyos elemeit színezi. A listában csak a befestendő mezők sorszámát adjuk meg. A bal alsó mező az 1. sorszámot viseli.



kép 3 2 15 [1 5]



kép 2 3 30 [1 3 5]



kép 4 3 15 [1 3 5 8 10 12]

```

eljárás kép :sor :oszlop :méret :lista
  ismétlés :sor [ismétlés :oszlop [ismétlés 4 [előre :méret jobbra 90]
                                jobbra 90 előre :méret balra 90]
                                jobbra 90 hátra :oszlop*:méret balra 90 előre :méret]
  hátra :sor*:méret töltőszín! "piros festi :oszlop :méret :lista
vége
  
```

A lista alapján kifesti a szükséges négyzeteket. A cella helyét meghatározhatjuk úgy, hogy a sorszá-  
mot elosztjuk az oszlopszámmal. Az eredmény lesz a sor, ahová rajzolni kell, a maradék pedig az  
oszlop száma mínusz egy.

```

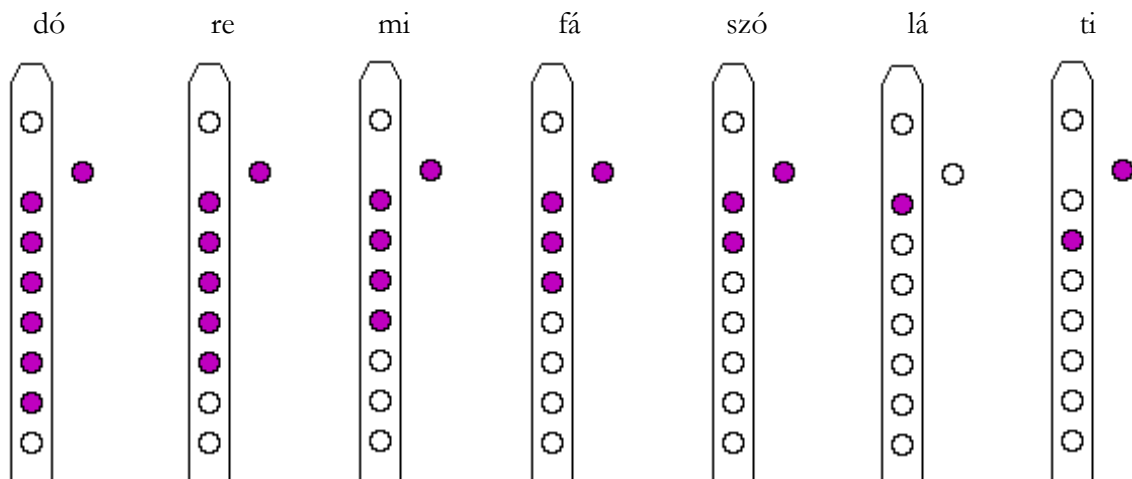
eljárás festi :oszlop :méret :lista
  ha nem üres? :lista [
    tollatfel jobbra 45 előre :méret/2 balra 45
    előre :méret*egészhányados (első :lista)-1 :oszlop jobbra 90
    előre :méret*mod (első :lista) - 1 :oszlop tölt
    hátra :méret* mod (első :lista) - 1 :oszlop balra 90
    hátra :méret* egészhányados (első :lista )-1 :oszlop
    jobbra 45 hátra :méret/2 balra 45 tollatle
    festi :oszlop :méret elsőnélküli :lista
  ]
vége
  
```

## Vezérlés szavak listájával

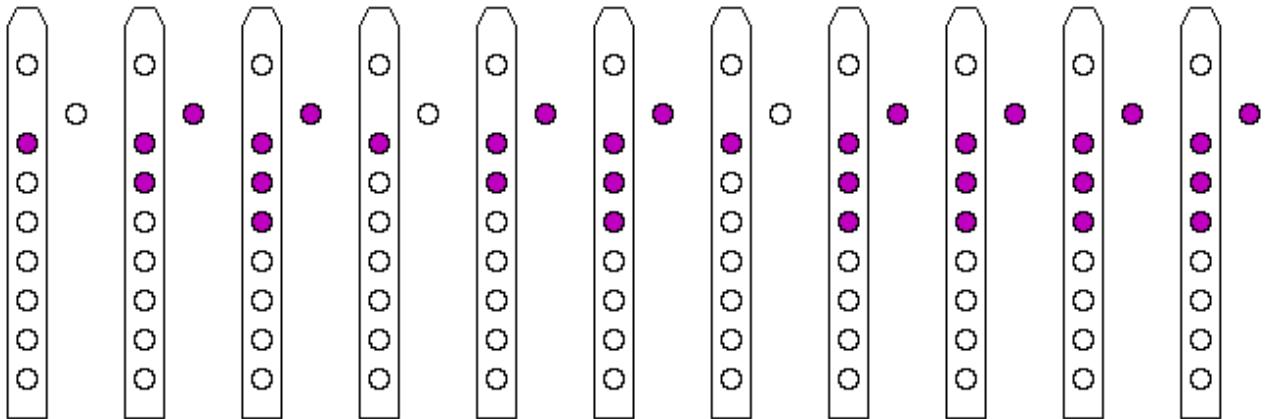
### Nóta, pozíció vezérlése újabb lista beiktatásával

Készíts Logo eljárást (nóta :dallam), amely egy dallam lejátszásához kirajzolja, hogy a furulyán  
milyen lyukakat kell befogni! Az adott hang megszólaltatásához befogott lyukak lila színűek, a töb-  
biek pedig üresek. A furulya hátoldalán is van egy lyuk, ennek befogását a furulya mellé rajzolt kör  
mutatja. Készítsd el az egyes hangok furulyafogását kirajzoló eljárásokat (dó, re, mi, fá,  
szó, lá, ti) is!

Az egyes hangok:



A nóta [lá szó fá lá szó fá lá fá fá fá fá] az alábbi furulya-sorozatot rajzolja:



A megadott dallam szerint hangonként kirajzoljuk a furulya lefogásait.

```

eljárás nóta :dallam
  ha nem üres? :dallam [hangonként első :dallam
    tollatfel jobbra 90 előre 60 balra 90 tollatle
    nóta elsónélküli :dallam
    tollatfel jobbra 90 hátra 60 balra 90 tollatle]
vége
  
```

Az egyes hangokhoz megadjuk, hogy melyik lyukakat kell majd lefogni.

```

eljárás hangonként :hang
  tollatle furulya tollatfel
  ha :hang="dó" [lefog [2 3 4 5 6 7 8]]
  ha :hang="ré" [lefog [3 4 5 6 7 8]]
  ha :hang="mi" [lefog [4 5 6 7 8]]
  ha :hang="fá" [lefog [ 5 6 7 8]]
  ha :hang="szó" [lefog [6 7 8]]
  ha :hang="lá" [lefog [7 ]]
  ha :hang="ti" [lefog [6 8]]
  tollatle
vége
  
```

Kirajzoljuk a furulyát.

```

eljárás furulya
  tollatle előre 200 jobbra 30 előre 10 jobbra 60 előre 10
  jobbra 60 előre 10 jobbra 30 előre 200 jobbra 90 előre 20 hátra 5
  jobbra 90 tollatfel jobbra 90 előre 5 balra 90
  ismétlés 7 [előre 20 tollatle kör 10 tollatfel]
  előre 20 jobbra 90 előre 30 tollatle kör 10 tollatfel
  hátra 30 balra 90
  előre 20 tollatle kör 10 tollatfel
  hátra 180 jobbra 90 hátra 10 balra 90 tollatle
vége
  
```

A lefogandó lyukak listája alapján végigmegyünk a lyukakon és beszínezzük őket.

```

eljárás lefog :lista
  ha nem üres? :lista
    [jobbra 90 előre 10 balra 90 előre 20
     előre ((első :lista)-1)*20
     ha első :lista =8 [jobbra 90 előre 30 balra 90 ]
     tölt
     hátra ((első :lista)-1)*20
     ha első :lista =8 [jobbra 90 hátra 30 balra 90]
     hátra 20 jobbra 90 hátra 10 balra 90
     lefog elsőnélküli :lista
    ]
vége

```

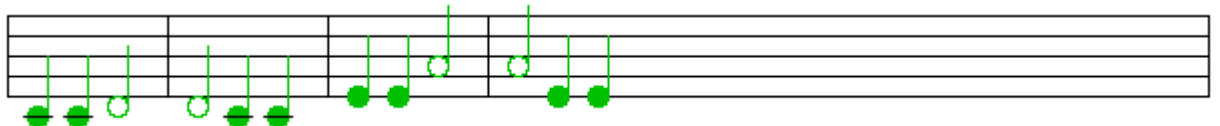
## Vezérlés vegyes listával

### Kotta, alak és pozíció vezérlése

Egy egyszerű kotta egy téglalappal adható meg, öt, egymástól egyenlő távolságra levő párhuzamos vonalat a végein összekötöttünk. A kottára hangjegyeket (cé, dé, é, ef, gé, á, há) rajzolhatunk. A különböző hosszúságú hangjegyekhez különböző ábra tartozik. A negyedhangokat teli, a félhangokat üres körrel, az egész hangokat pedig szár nélküli üres körrel rajzoljuk. Minden negyedik negyedhang után a következő hangot egy függőleges vonal választja el az előzőtől.



Készíts Logo eljárást (zenél :hangjegyek), amely kirajzolja a paraméterként megadott hangjegyeket a kottára (biztosan ráfér)! A hangjegyek között elhelyezhetünk számokat – 1,2,4 – ami azt jelenti, hogy ettől kezdve egész, fél, illetve negyedhangok következnek. (Ha csak angol betűket használhatsz eljárásnévnek, akkor a cee, dee, ee, ef, gee, aa, haa neveket használj!)



zenél [cé cé 2 dé dé 4 cé cé é é 2 á á 4 é é]

Számokból és szavakból álló listák feldolgozásánál fel kell ismernünk, hogy a feldolgozandó elem szám-e – ezt a szám? függvény meghívásával lehet meghatározni .

```

eljárás zene :z :h :t
  ha nem üres? :z
    [hakülönben szám? első :z [zene elsőnélküli :z első :z :t]
     [játsszik első :z :h :t zene elsőnélküli :z :h :t+4/:h]]
vége

```

```

eljárás játszik :hang :h :t
  ha és :t > 0 0 = maradék :t 4 [tollszín! 0 szünet előre 40 hátra 40]
  ha :hang="cé [szünet cé :h]
  ha :hang="dé [szünet dé :h]
  ha :hang="é [szünet é :h]
  ha :hang="ef [szünet ef :h]
  ha :hang="gé [szünet gé :h]
  ha :hang="á [szünet á :h]
  ha :hang="há [szünet há :h]
vége

```



A kotta egy 300 egység szélességű, 10 egység magasságú téglalapokból álló alakzat.

```
eljárás kotta
  tollszín! 0
  ismétlés 4 [jobbra 90 előre 600 hátra 600 balra 90 előre 10]
  jobbra 90 előre 600 balra 90 hátra 40 jobbra 90 hátra 600
  tollatfel hátra 20 tollatle balra 90
vége
```

Az egyes hangjegyek az előzőhöz képest 20 egységgel jobbra helyezkednek el.

```
eljárás szünet
  tollatfel jobbra 90 előre 20 balra 90 tollatle
vége
```

Az á hang a kotta aljától számítva 15 egység távolságra van felfelé, a c pedig 10 egységgel lefelé, s van rajta egy vízszintes áthúzás.

```
eljárás á :h
  tollatfel előre 15 tollatle hangjegy :h
  tollatfel hátra 15 tollatle
vége
```

```
eljárás cé :h
  tollatfel hátra 10 tollatle hangjegy :h jobbra 90
  tollszín! 0 hátra 12 előre 14 hátra 2 balra 90
  tollatfel előre 10 tollatle
vége
```

Az egyes hangjegyek 10 egység átmérőjű körlapok, amelyeknek 30 egység hosszú szára van jobboldalt. Ehhez hasonlóan lehet a többi hangot is megrajzolni a hangjegy segítségével. Az egész hangnak nincs jobboldali szára, a többinek van. A negyed hangot be kell festeni, a többit nem.

```
eljárás hangjegy :h
  tollszín! 2 töltőmód! 1
  ismétlés 360 [előre 31,4159/360 balra 1]
  ha :h=4 [tollatfel balra 60 előre 2 tölt hátra 2 jobbra 60]
  ha :h>1 [tollatle előre 30 hátra 30]
vége
```

```
eljárás zenél :z
  törölkép kotta szünet zene :z 4 0
vége
```

### Parkolás, színek és pozíció vezérlése

Készítsd el az alábbi autót (autó :méret :szín) és parkolót rajzoló (parkolás :db :méret :leírás) eljárásokat, ahol a kocsi méretét a :méret, színét pedig a :szín paraméter határozza meg! A parkoló elkészítésénél a kocsi helyek számát a :db, egy helyének hosszát a :méret, a beálló kocsikat a megjelenítendő színekkel a :leírás listában adjuk meg.



autó 100 4



parkolás 5 80 [1 2 nincs 4]

Jobb, ha részekre bontjuk a rajzot (az autó kerekekből és karosszériából áll) és ezeket külön-külön rajzoljuk ki.

```
eljárás autó :méret :szín
  tollatfel jobbra 90 előre :méret/5 balra 90 tollatle
  kerék :méret/10 tollatfel jobbra 90 előre :méret/5*3
  balra 90 tollatle
  kerék :méret/10 tollatfel jobbra 90 hátra :méret*4/5
  balra 90 tollatle karosszéria :méret :szín
vége

eljárás kerék :sugár
  ; kör közepéből indul
  tollatfel előre :sugár jobbra 90 tollatle
  ismétlés 360 [előre 2*:sugár*3,14/360 jobbra 1]
  jobbra 90 tollatfel előre :sugár
  töltőszín! 0 tölt jobbra 180 tollatle
vége

eljárás karosszéria :méret :szín
  töltőszín! :szín
  tollatle előre :méret/5 jobbra 90 előre :méret/5
  balra 60 előre :méret/5 jobbra 60 előre :méret/5*2
  jobbra 60 előre :méret/5 balra 60 előre :méret/5
  jobbra 90 előre :méret/5 jobbra 90 előre :méret jobbra 90
  tollatfel jobbra 45 előre :méret/5 tölt hátra :méret/5
  balra 45 tollatle
vége

eljárás parkolás :db :méret
  parkoló :db :méret
  beáll :méret :autók
vége

eljárás parkoló :db :méret
  ismétlés :db [előre :méret/2 jobbra 90 előre :méret
                jobbra 90 előre :méret/2 jobbra 180]
  tollatfel jobbra 90 hátra :db*:méret balra 90
vége
```

Az autók listában megadottak szerint rajzoljuk ki a beálló autókat. Ha a „nincs szót találjuk, vagy vége a listának, akkor nem rajzolunk autót az adott helyre.

```
eljárás beáll :méret :autók
  ha nem üres? :autók [hakülönben nem egyenlő? első :autók "nincs
                        [autó :méret első :autók] []
                        tollatfel jobbra 90 előre :méret
                        balra 90 tollatle
                        beáll :méret elsőnélküli :autók
                        tollatfel jobbra 90 hátra :méret
                        balra 90 tollatle]
vége
```

## Vezérlés két listával

### Parkolás, 2. verzió

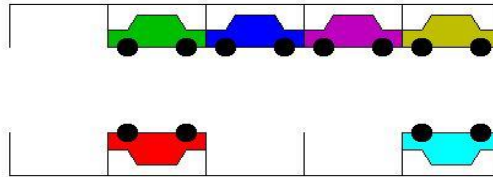
Egy parkolóban az autósok inkább az egyik oldalra szeretnek parkolni (az üzletek közelében), de ha a korábban kinézett hely már foglalt, akkor azzal szemben állnak le.

Készíts egy autót (autó :méret :szín) és a parkolást megjelenítő eljárásokat (parkolás :db :méret :foglalt :újautók)! A :méret egy autó helyének hosszát, a :szín az

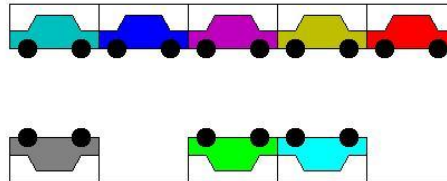
autó színét, a :db oldalanként az autóhelyek számát, a : foglalt a korábban már parkolt kocsik listáját, az : újautók pedig az újonnan érkezők listáját tartalmazza, hely szerint felsorolva.



autó 100 5



parkolás 5 80 [nincs 2 nincs 5 6] [nincs 12 9 nincs 11]



parkolás 5 80 [3 nincs 5 6][8 9 10 11 12]

```
eljárás parkolás :db :méret :autók :autók2
  parkoló :db :méret
  tollatfel jobbra 90 előre :méret*:db jobbra 90 előre :méret tollatle
  parkoló :db :méret
  tollatfel hátra :méret balra 90 hátra :méret*:db balra 90 tollatle
  beáll :méret :autók :autók2
vége
```

Az autók listákban megadottak szerint rajzoljuk ki a beálló autókat a parkoló két oldalára. Ha a „nincs szót találjuk, vagy vége a listának, akkor nem rajzolunk autót az adott helyre.

```
eljárás beáll :méret :autók :autók2
  ha nem üres? :autók
    [hakülönben és nem egyenlő? első :autók "nincs egyenlő? első
      :autók2 "nincs [autó :méret első :autók][
    hakülönben és egyenlő? első :autók "nincs nem egyenlő? első
      :autók2 "nincs [autó :méret első :autók2][
    hakülönben és nem egyenlő? első :autók "nincs nem egyenlő? első
      :autók2 "nincs [autó :méret első :autók tollatfel
        jobbra 90 előre :méret jobbra 90
        előre :méret tollatle
        autó :méret első :autók2 tollatfel
        hátra :méret balra 90 hátra :méret
        balra 90][
  tollatfel jobbra 90 előre :méret balra 90 tollatle
  beáll :méret elsőnélküli :autók elsőnélküli :autók2
  tollatfel jobbra 90 hátra :méret balra 90 tollatle]
vége
```

Jobb, ha részekre bontjuk a rajzot és ezeket külön-külön rajzoljuk ki.

```
eljárás autó :méret :szín
  tollatfel jobbra 90 előre :méret/5 balra 90 tollatle
  kerék :méret/10 tollatfel jobbra 90 előre :méret/5*3
  balra 90 tollatle
  kerék :méret/10 tollatfel jobbra 90 hátra :méret*4/5
  balra 90 tollatle karosszéria :méret :szín
vége
```

```
eljárás kerék :sugár
  tollatfel előre :sugár jobbra 90 tollatle
  ismétlés 360 [előre 2*:sugár*3,14/360 jobbra 1]
  jobbra 90 tollatfel előre :sugár
  töltőszín! 0 tölt jobbra 180 tollatle
vége
```

```
eljárás karosszéria :méret :szín
  töltőszín! :szín
  tollatle előre :méret/5 jobbra 90 előre :méret/5
  balra 60 előre :méret/5 jobbra 60 előre :méret/5*2
  jobbra 60 előre :méret/5 balra 60 előre :méret/5
  jobbra 90 előre :méret/5 jobbra 90 előre :méret
  jobbra 90
  tollatfel jobbra 45 előre :méret/5 tölt hátra :méret/5
  balra 45 tollatle
vége
```

## Vezérlés listák listájával

### Rovásírás, alak kirajzolása

A rovásírás betűi egyenes vonalakkból állnak. Készíts Logo eljárást (rovás :mondat), amely az alábbi betűket tartalmazó mondatok szavait tudja a képernyőre rajzolni!

4 : a   4 : á   † : i   † : í   3 : m   3 : p   H : r   Y : t

rovás [már itt a tapír]   HT 3 4 Y 4 Y Y † H 4 3

A mondatokat szavakból, a szavakat karakterekből építjük fel. Listák listájáról van tehát szó! Figyeljünk rá, hogy a rovásírás iránya jobbról balra halad, így a feldolgozást is a mondat végéről kezdjük!

```
eljárás rovás :mondat
  ha nem üres? :mondat [rovásb utolsó :mondat tollatfel jobbra 90
    előre 40 balra 90 tollatle
    rovás utolsónélküli :mondat]
vége
```

```
eljárás rovásb :szó
  ha nem üres? :szó [betű első :szó 100 rovásb elsónélküli :szó]
vége
```

```
eljárás abetu :h
  előre :h balra 135 előre gyök 2*:h*:h/9 balra 135
  előre :h/3 balra 90 hátra 2*:h/3
vége
```

```
eljárás aabetu :h
  előre :h ismétlés 2 [balra 120 előre :h/2]
  balra 120 hátra :h/2
vége
```

```
eljárás ibetu :h
  előre :h hátra :h/4 balra 105 előre :h/4 hátra :h/2 előre :h/4
  jobbra 105 hátra 3*:h/4
vége
```

```
eljárás iibetu :h
  előre 9*:h/10 balra 105 előre :h/4 hátra :h/2 előre :h/4
  jobbra 105 hátra 9*:h/10
vége

eljárás mbetu :h
  előre :h ismétlés 2 [balra 120 előre :h/2]
  jobbra 120 ismétlés 2 [előre :h/2 balra 120]
vége

eljárás pbetu :h
  előre :h balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3 balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3 balra 120 előre 2*:h/3 hátra 2*:h/3 jobbra 120
  hátra :h/3
vége

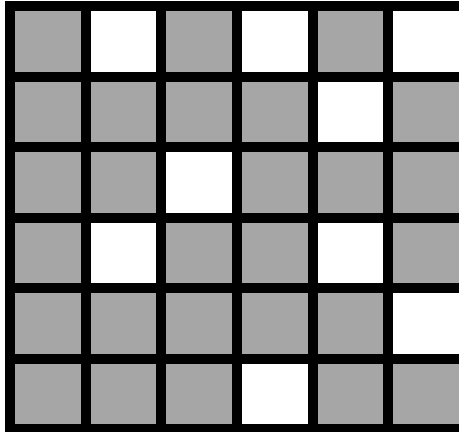
eljárás rbetu :h
  előre :h hátra 3*:h/5 jobbra 66 előre :h/2 balra 66
  előre 2*:h/5 hátra :h előre 3*:h/5 jobbra 66 hátra :h/2
  balra 66 hátra 2*:h/5
vége

eljárás tbetu :h
  előre :h hátra :h/3 balra 45 előre gyök 2*:h*:h/9
  hátra gyök 2*:h*:h/9 jobbra 45 hátra 2*:h/3
vége

eljárás betű :b :h
  ha :b="a" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    abetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="á" [tollatfel jobbra 90 előre :h/2 balra 90 tollatle
    aabetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="i" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    ibetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="í" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    iibetu :h
    tollatfel jobbra 90 előre :h/3 balra 90 tollatle]
  ha :b="m" [tollatfel jobbra 90 előre :h/2 balra 90 tollatle
    mbetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="p" [tollatfel jobbra 90 előre 3*:h/5 balra 90 tollatle
    pbetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
  ha :b="r" [tollatfel jobbra 90 előre :h/10 balra 90 tollatle
    rbetu :h
    tollatfel jobbra 90 előre :h/2 balra 90 tollatle]
  ha :b="t" [tollatfel jobbra 90 előre :h/3 balra 90 tollatle
    tbetu :h
    tollatfel jobbra 90 előre :h/10 balra 90 tollatle]
vége
```

## Titkosírás, pozíció vezérlése

Verne Gyula Sándor Mátyás című regényében egy olyan titkosírást használnak, amelyben egy 6x6-os négyzetrácsos lap bizonyos mezőit kivágják (ezt hívják rostélynek), majd egy ugyanilyen négyzetes elrendezésben szereplő betűtáblázat fölé helyezve a látható betűket folyamatosan kiolvassák.



Készíts Logo eljárást `rostély` :paraméter, amely egy pozícióival megadott rostélyt kirajzol!

Az ábrán látható rostélyt az alábbi paraméterekre kapjuk:

```
rostély [[2 4 6] [5] [3] [2 5] [6] [4]]
```

Nem kell mást tennünk, mint egy mozaikot rajzolni, amelyben az egyes cellák töltő színét határozzuk meg a listával.

```
eljárás rostély :ly
  tollvastagság! 3 mozaik 30 1 :ly
vége

eljárás mozaik :h :s :ly
  sor :h 1 utolsó :ly
  ha :s<6 [előre :h mozaik :h :s+1 utolsónélküli :ly
          tollatfel hátra :h tollatle]
vége

eljárás sor :h :s :ly
  hakülönben :s<7 [ha üres? :ly [alap :h 7 sor :h :s+1 :ly]
                  [hakülönben :s<első :ly [alap :h 7 sor :h :s+1 :ly]
                    [alap :h 15 sor :h :s+1 elsőnélküli :ly]]
                  jobbra 90 hátra :h balra 90]
vége

eljárás alap :h :sz
  ismétlés 4 [előre :h jobbra 90]
  töltőszín! :sz
  tollatfel jobbra 45 előre :h tölt hátra :h balra 45 tollatle
  jobbra 90 előre :h balra 90
vége
```

## Mozaik

Készíts négyzetes mozaikot (mozaik :méret :leírás) úgy, hogy az egyes elemek helyére képeket „nyomtatunk”. Az, hogy hova milyen kép kerüljön egy összetett listával adjuk meg. Több különböző lehetőség között választhatunk: 0-üresen hagyjuk az adott helyet, számoknál – a sorszámozott kép lenyomatát készítjük el. Tetszőleges képet használhatunk!



eljárás leírás

```
eredmény [[ 5 0 0 2 ] [ 0 0 0 0 ] [ 1 0 0 3 ] [ 4 4 4 4 ] [ 4 4 4 4 ] ]  
vége
```

eljárás mozaik :méret :leírás

```
ha nem üres? :lista [kirajzolsor :méret első :leírás  
tollatfel előre :méret tollatle  
mozaik :méret elsőnélküli :leírás  
tollatfel hátra :méret tollatle ]
```

vége

eljárás kirajzolsor :méret :lista

```
ha nem üres? :lista [  
ismétlés 4 [előre :méret jobbra 90 ]  
ha első :lista <> 0 [alak! szó első :lista ".jpg lenyomat]  
jobbra 90 előre :méret balra 90  
kirajzolsor :méret elsőnélküli :lista  
jobbra 90 hátra :méret balra 90 ]
```

vége

## Szövegmanipuláció

Első hallásra, ha egy „szövegre” gondolunk, úgy vélhetjük, hogy az nem lehet túl érdekes a programozás, adattárolás szempontjából. De koránt sincs így! Gondoljunk csak egy HTML oldal forráskódjára, ami nem más, mint egy egyszerű szöveg. Egy HTML oldalon belül azonban tetszőleges mélységben lehetnek HTML elemek egymásba ágyazva! A szövegben a benne levő jelzések, tagek segítségével hierarchikus adatok is tárolhatóak! Összetett adatok tárolására a HTML nyelvhez hasonló, de annál általánosabb XML formátumot használják, de a JavaScript-es világban elterjedt a JSON formátumú adattárolás is. Az XML formátumról a <http://xml.lap.hu> címen olvashatunk többet. Ma nagyobb adatmennyiség tárolása esetén is egyre többször választják a szöveges adatbázisokat, mert nagy előnyük a könnyen változtatható, laza szerkezetük.

A szövegkezeléssel kapcsolatos feladatok tehát igencsak fontosak az adattárolással kapcsolatos tevékenységek során! Érdemes megismerni ennek a területnek a lehetőségeit a Logo környezetben is. A Logo-ban a szövegkezelés és a listakezelés egy tőről fakad, hiszen a szót a karakterek listájának, a mondatot a szavak listájának, a bekezdést a mondatok listájának stb. tekinthetjük! Vizsgáljuk meg ezeket közelebbről is!

### A feladat eredménye lista

#### Bumm játék

A Bumm nevezetű társasági játékban a játékosok egy kiinduló számtól kezdve egyesével mondják a következő számot, DE: van egy tiltott szám, s ha olyan szám következne, amelyiknek valamelyik számjegye éppen a tiltott szám, vagy a kimondandó szám osztható a tiltott számmal, akkor a szám helyett azt kell mondani, hogy „Bumm”. Aki eltéveszti, zálogot ad.

Írd meg a bumm :ettől :eddig :tiltott eljárást, amely kiírja a játékban elhangzottakat! A program a „Bumm” helyett két felkiáltójelet írjon!

```
bumm 7 14 3 eredménye: 7 8 !! 10 11 !! !! 14
```

Végighalad a megadott számintervallumon és az eredménylistába írja az aktuális számot vagy a bummot.

```
eljárás bumm :honnan :meddig :mi
  ha :honnan>:meddig [eredmény []]
  eredmény elsőnek érték :honnan :mi bumm :honnan+1 :meddig :mi
vége
```

Az adott számról a :miben eldönti, hogy benne van-e a tiltott szám (:mi), vagy osztható-e vele és az eredményt ennek hatására képzí.

```
eljárás érték :miben :mi
  ha (maradék :miben :mi)=0 [eredmény "!!]
  hakülönben vanbenne :miben :mi [eredmény "!!][eredmény :miben]
vége
```

```
eljárás vanbenne :miben :mi
  ha üres? :miben [eredmény "hamis]
  ha :mi=első :miben [eredmény "igaz]
  eredmény vanbenne elsőnélküli :miben :mi
vége
```



## A feladat bemenete és eredménye is szó

### Titkosírás

Egy titkosírást úgy készítenek, hogy egy táblázat alapján az ábécé betűit más betűkre cserélik. Ké- szíts, függvényeket kódol :szó, megfejt :szó, amelyek az alábbi 2 konstansfüggvény segítségével megadja a paraméterként megadott szó kódját, illetve a kódolt szó eredeti alakját! A megoldásban csak kisbetűkkel kell foglalkozni!

```
kódol "alma eredménye öőüö
megfejt "whwk eredménye eper
eljárás kódol :szó
  ha üres? :szó [eredmény " ]
  eredmény elsőnek helyettesít első :szó ábécé kódja
  kódol elsőnélküli :szó
vége
eljárás megfejt :szó
  ha üres? :szó [eredmény " ]
  eredmény elsőnek helyettesít első :szó kódja ábécé
  megfejt elsőnélküli :szó
vége
eljárás helyettesít :b :ábécé :kódja
  ha üres? :ábécé [eredmény :b]
  ha :b=első :ábécé [eredmény első :kódja]
  eredmény helyettesít :b elsőnélküli :ábécé elsőnélküli :kódja
vége
eljárás kódja
  eredmény "öüöüqwertzuiopőúasdfghjkléáíyxcvbnm
vége
eljárás ábécé
  eredmény "aábcdeéfgghiijklmnoóöőpqrstuúüúvwxyz
vége
```

### Akasztófa játék

Az akasztófa játékban egy ismeretlen szót kell kitalálni. A játék kezdetén a kitalálendő szó minden betűje helyett egy-egy pontot írunk. Ezután a játékos mond egy betűt. Ha ez benne van a szóban, akkor az előfordulási helyére kiírjuk, különben pedig egy vonalat húzunk az akasztófához. A játékos nyer, ha hamarabb kitalálja a szót, mint ahogy elkészül az akasztófa.

Készíts függvényt (akasztófa :kitalálendő :mostholtart :betű), amely az akasztófa játék egy lépését szimulálja! Eredményként adja vissza a módosult szöveget, ha a betű benne van a szóban! Vegye észre, ha kitaláltuk a szót és ekkor tegyen a végére egy felkiáltójelet! Ha a betű nincs a szóban vagy már korábban kitaláltuk, akkor a szót változatlanul adja meg!

Ha a kitalálendő szó az ALMAFA és most kezdjük a játékot, akkor

```
akasztófa "ALMAFA "..... "A      eredménye "A..A.A
akasztófa "ALMAFA "A..A.A "F      eredménye "A..AFA
akasztófa "ALMAFA "A.MAFA "L      eredménye "ALMAFA!
```

```
eljárás akasztófa :kitalálándó :mostholtart :betű
  ha üres? :kitalálándó [eredmény "]
  ha :betű=első :kitalálándó
    [eredmény elsőnek :betű akasztófa elsőnélküli :kitalálándó
      elsőnélküli :mostholtart :betű]
  eredmény elsőnek első :mostholtart
    akasztófa elsőnélküli :kitalálándó
      elsőnélküli :mostholtart :betű
vége
```

### A feladatnak két szó a bemenete

#### Anagramma

Két szó anagrammája egymásnak, ha ugyanazokból a betűkből állnak, csak esetleg más sorrendben (pl. ALFA és FALA). Írj Logo függvényt (anagramma :egyik :másik), amely eldönti két szóról, hogy anagrammák-e!

Az egyik szót az elejéről, a másikat a végétől vizsgáljuk karakterenként.

```
eljárás anagramma :egyik :másik
  ha és üres? :egyik üres? :másik [eredmény "igaz]
  ha üres? :egyik [eredmény "hamis]
  ha üres? :másik [eredmény "hamis]
  ha első :egyik <> utolsó :másik [eredmény "hamis]
  eredmény anagramma elsőnélküli :egyik utolsónélküli :másik
vége
```

### A feladatnak két lista a bemenete

#### Kártyázás

A 21-es kártyajátékot magyar kártyával játsszák. Először mindenki kap két lapot, majd felváltva húznak, amíg kérnek. A lapok értéke 2,3,4,7,8,9,10,11 lehet.

Készíts Logo függvényt (kinyert :pakli :húzások), amely egy 21-est játszó párosnál megadja, hogy ki nyert! A :pakli listában a kártyák értékeit találjuk. A :húzások listában az egymás után következő húzásokhoz tartozó személyek azonosítója áll. [A A B B] azt jelenti, hogy 2 alkalommal A kapott lapot a pakliból, majd kettőt B is kap. Ha valaki lapjainak az összege 21-nél több, az veszett („befuccsolt”), egyébként az nyer, aki lapjainak nagyobb az értéke.

```
Kinyert [2 3 7 2 10 3 4 ] [A A B B A B]
```

```
eredménye A lapjai: 15, B lapjai: 12, tehát A nyert.
```

Összeszámoljuk a két játékos által kapott lapok pontjait és meghívjuk a kiértékelést végző nyeres eljárást.

```
eljárás kinyert :pakli :húzások
  nyeres összeszámol "A :pakli :húzások
    összeszámol "B :pakli :húzások
vége
```

A paraméterben megadott személy pontjait számoljuk össze a megkapott húzások és pakli alapján.

```
eljárás összeszámol :ki :pakli :húzások
  ha üres? :húzások [eredmény 0]
  ha első :húzások=:ki [eredmény (első :pakli)+
    összeszámol :ki elsőnélküli :pakli elsőnélküli :húzások]
  eredmény összeszámol :ki elsőnélküli :pakli elsőnélküli :húzások
vége
```

A paraméterként kapott pontszámok alapján kiértékeljük a játszmát.

```

eljárás nyerés :A :B
  ha és (:A>21) (:B>21) [ki [befuccsoltak mind a ketten]]
  ha és (:A>21) (:B<=21) [ki [nyert a B]]
  ha és (:A<=21) (:B>21) [ki [nyert a A]]
  ha és és (:A<=21) (:B<=21) (:A<:B) [ki [nyert a B]]
  ha és és (:A<=21) (:B<=21) (:A>:B) [ki [nyert az A]]
  ha és és (:A<=21) (:B<=21) (:A=:B) [ki [döntetlen]]
vége

```

## Kártya

A 21-es kártyajátékot magyar kártyával játsszák. Először mindenki kap két lapot, majd felváltva húznak, amíg kérnek. A lapok értéke 2,3,4,7,8,9,10,11 lehet, a figurák zöld, szív, makk és tők.

Készíts Logo függvényt (`kinyert :pakli :húzások`), amely egy 21-est játszó párosnál megadja, hogy ki nyert. A `:pakli` listában a kártyák értékeit találjuk. A `:húzások` listában az egymás után következő húzásokhoz tartozó személyek azonosítója áll. Az `[A A B B]` azt jelenti, hogy 2 alkalommal A kapott lapot a pakliból, majd kettőt B is kap. Ha valaki lapjainak az összege 21-nél több, az vesztett („befuccsolt”), egyébként az nyer, aki lapjainak nagyobb az értéke.

```

kinyert [M2 Z3 T7 S2 S10 M3 T4 ] [A A B B A B]
eredménye A lapjainak értéke: 15, B lapjainak értéke: 12,
tehát A nyert.

```

Összeszámoljuk a két játékos által kapott lapok pontjait és meghívjuk a kiértékelést végző nyerés eljárást.

```

eljárás kinyert :pakli :húzások
  nyerés összeszámol "A :pakli :húzások
  összeszámol "B :pakli :húzások
vége

```

A paraméterben megadott személy pontjait számoljuk össze a megkapott húzások és pakli alapján. A lap értékét a szín „eldobásával” kaphatjuk meg.

```

eljárás összeszámol :ki :pakli :húzások
  ha üres? :húzások [eredmény 0]
  ha első :húzások=:ki [eredmény (elsőnélküli első :pakli)+
    összeszámol :ki elsőnélküli :pakli elsőnélküli :húzások]
  eredmény összeszámol :ki elsőnélküli :pakli elsőnélküli :húzások
vége

```

A paraméterként kapott pontszámok alapján kiértékeljük a játszmát.

```

eljárás nyerés :A :B
  ha és (:A>21) (:B>21) [ki [befuccsoltak mind a ketten]]
  ha és (:A>21) (:B<=21) [ki [nyert a B]]
  ha és (:A<=21) (:B>21) [ki [nyert a A]]
  ha és és (:A<=21) (:B<=21) (:A<:B) [ki [nyert a B]]
  ha és és (:A<=21) (:B<=21) (:A>:B) [ki [nyert az A]]
  ha és és (:A<=21) (:B<=21) (:A=:B) [ki [döntetlen]]
vége

```

## A feladat bemenete és eredménye is mondat/lista

### Ékezetek

Készíts Logo függvényt (szűr :mondat), amely egy magyar mondatból előállít egy olyan mondatot, amelyben az eredeti mondat ékezetes betűket tartalmazó szavait annyi \* karakterrel helyettesíti, ahány betűből álltak! A megoldás részeként készíts egy függvényt (ékezetes? :szó), amely egy tetszőleges szóról megmondja, hogy tartalmaz-e ékezetes betűket!

```
szűr [ez is jó hír lesz]      ⇒ [ez is ** *** lesz]
ékezetes? "ez                 ⇒ "hamis
ékezetes? "jó                 ⇒ "igaz
```

Vizsgáljuk végig szavanként, hogy van-e az adott szóban ékezetes betű! Készítsünk egy segéd eljárást, amely a paraméterben megadott számú csillagot fűz össze!

```
eljárás szűr :mondat
  ha üres? :mondat [eredmény :mondat]
  ha ékezetes? első :mondat
    [eredmény elsőnek csillagok elemszám első :mondat
      szűr elsőnélküli :mondat]
  eredmény elsőnek első :mondat szűr elsőnélküli :mondat
vége

eljárás csillagok :db
  ha :db=0 [eredmény "]
  eredmény elsőnek "*" csillagok :db-1
vége

eljárás ékezetes? :szó
  ha üres? :szó [eredmény "hamis]
  ha eleme? első :szó ékezetesbetűk [eredmény "igaz]
  eredmény ékezetes? elsőnélküli :szó
vége
```

Az ékezetes betűket egy konstans függvénnyel adjuk meg!

```
eljárás ékezetesbetűk
  eredmény "áéíóöőüűúÁÉÍÖÖÜŰ
vége
```

### Számok

Egy mondatban szavak és pozitív egész számok is szerepelnek. Készíts Logo eljárást (átalakít :mondat), amely a mondatban a 100-nál kisebb számokat lecseréli betűkkel írtakra, a nagyobbakat pedig változatlanul hagyja!

```
átalakít [A kertben 26 kutya és 4 macska játszott 100
  egérrel.] ⇒
[A kertben huszonhat kutya és négy macska játszott 100
  egérrel.]
```

Szavanként dolgozzuk fel a mondatot, ha számot találunk, helyette a neki megfelelő szöveges formát írunk!

```
eljárás átalakít :mondat
  ha üres? :mondat [eredmény :mondat]
  ha és szám? első :mondat elemszám első :mondat <=2
    [eredmény elsőnek szöveggé első :mondat
      átalakít elsőnélküli :mondat]
  eredmény elsőnek első :mondat átalakít elsőnélküli :mondat
vége
```

Az átalakítás során figyeljünk a kivételes esetekre (10,20)!

```
eljárás szöveggé :szám
  ha 1=elemszám :szám [eredmény elem 1+:szám egyes]
  ha 2<elemszám :szám [eredmény :szám]
  ha :szám="10 [eredmény "tíz]
  ha :szám="20 [eredmény "húsz]
  ha 0=utolsó :szám [eredmény elem első :szám tizes]
  eredmény szó elem első :szám tizes elem 1+utolsó :szám egyes
vége
```

```
eljárás tizes
  eredmény [tizen huszon harminc negyven ötven hatvan hetven
            nyolcvan kilencven]
vége
```

```
eljárás egyes
  eredmény [nulla egy kettő három négy öt hat hét nyolc kilenc]
vége
```

### Ezresek

Egy szövegben vegyesen fordulnak elő számok (1 és 1 milliárd között) és szavak. A számokat mindig számjegyekkel írták. Írj Logo eljárást (szöveggé :mondat), amely az **ezer**, illetve **millió** szót a számokba illeszti!

```
szöveggé [1000000 nyúl és 30000 róka kergetőzött 2027 ré-
ten.]
eredménye: 1 millió nyúl és 30 ezer róka kergetőzött 2
ezer 27 réten.
```

A számok közül az ezerre, illetve a millióra végződőkben a nullákat az **ezer**, illetve a millió szóval írták. Írj Logo eljárást (számmá :mondat), amely az ezerre, illetve millióra végződőkben az ezer, illetve a millió szavakat három, illetve hat darab nullára cseréli!

```
számmá [1 millió nyúl és 30 ezer róka kergetőzött 2 ezer
réten.]
eredménye: 1000000 nyúl és 30000 róka kergetőzött 2000
réten.
```

```
eljárás szöveggé :mondat
  ha üres? :mondat [eredmény []]
  ha szám? első :mondat
    [eredmény mondat átalakít első :mondat
      szöveggé elsőnélküli :mondat]
  eredmény mondat első :mondat szöveggé elsőnélküli :mondat
vége
```

Az átalakításban a milliós, illetve ezres számokat két részből állítjuk össze, a számmal kiírt nagy számokká és a számokkal kiírt maradékká.

```
eljárás átalakít :szám
  ha :szám>=1000000
    [eredmény (mondat egészhányados :szám 1000000 "millió
      átalakít maradék :szám 1000000)]
  ha :szám>=1000
    [eredmény (mondat egészhányados :szám 1000 "ezer
      átalakít maradék :szám 1000)]
  ha :szám=0 [eredmény " ]
eredmény :szám
vége
```

A számmá alakításnál egyszerre két szót kell figyelni. Ha az első szám, akkor megvizsgáljuk, hogy a következő szó az „ezer” vagy a „millió”!

```
eljárás számmá :mondat
  ha vagy üres? :mondat üres? elsőnélküli :mondat
    [eredmény :mondat]
  ha és szám? első :mondat egyenlő? első elsőnélküli :mondat
    "millió
    [eredmény mondat 1000000*első :mondat
      számmá elsőnélküli elsőnélküli :mondat]
  ha és szám? első :mondat egyenlő? első elsőnélküli :mondat
    "ezer
    [eredmény mondat 1000*első :mondat
      számmá elsőnélküli elsőnélküli :mondat]
eredmény mondat első :mondat számmá elsőnélküli :mondat
vége
```

### Római számok

Egy történelmi szövegben római számok is szerepelhetnek. Készíts Logo eljárást (római :mondat), amely egy mondatban a római számokat arab számokra alakítja! A római szám számjegyei után vagy szóköz, vagy pont, vagy kötőjel szerepelhet. A római számok: I - 1, V - 5, X - 10, L - 50, C - 100, D - 500, M - 1000.

```
római [A XVIII. században MDCCXCII-ben történt.]
```

```
Eredménye: A 18. században 1792-ben történt.
```

A mondatot szavanként dolgozza fel.

```
eljárás római :mondat
  ha üres? :mondat [eredmény :mondat]
  ha római? első :mondat [eredmény elsőnek átalakít első :mondat
    római elsőnélküli :mondat]
eredmény elsőnek első :mondat római elsőnélküli :mondat
vége
```

Egy szót akkor tekint esetleg római számnak, ha az első karaktere római szám.

```
eljárás római? :szó
eredmény eleme? első :szó "IVXLCDM
vége
```

A római szám egy szó elején lehet, ha az utolsó karakter már nem római szám, akkor csak e nélkül próbálunk átalakítani.

```
eljárás átalakít :szó
  ha nem eleme? utolsó :szó "IVXLCDM
    [eredmény utolsónak utolsó :szó átalakít utolsónélküli :szó]
eredmény konvertál :szó
vége
```

Az egyes római számjegyek alapján a szám értékét határozza meg.

```

eljárás konvertál :szó
  ha üres? :szó [eredmény 0]
  ha "M=első :szó [eredmény 1000+konvertál elsőnélküli :szó]
  ha "D=első :szó [eredmény 500+konvertál elsőnélküli :szó]
  ha "C=:szó [eredmény 100]
  ha "C=első :szó
    [ha "M=első elsőnélküli :szó
      [eredmény 900+konvertál elsőnélküli elsőnélküli :szó]
      ha "D=első elsőnélküli :szó
        [eredmény 400+konvertál elsőnélküli elsőnélküli :szó]
        eredmény 100+konvertál elsőnélküli :szó]
    ha "L=első :szó [eredmény 50+konvertál elsőnélküli :szó]
  ha "X=:szó [eredmény 10]
  ha "X=első :szó
    [ha "C=első elsőnélküli :szó
      [eredmény 90+konvertál elsőnélküli elsőnélküli :szó]
      ha "L=első elsőnélküli :szó
        [eredmény 40+konvertál elsőnélküli elsőnélküli :szó]
        eredmény 10+konvertál elsőnélküli :szó]
    ha "V=első :szó [eredmény 5+konvertál elsőnélküli :szó]
  ha "I=:szó [eredmény 1]

  ha "I=első :szó
    [ha "X=első elsőnélküli :szó
      [eredmény 9+konvertál elsőnélküli elsőnélküli :szó]
      ha "V=első elsőnélküli :szó
        [eredmény 4+konvertál elsőnélküli elsőnélküli :szó]
        eredmény 1+konvertál elsőnélküli :szó]
vége

```

## Mondatok

Egy magyar mondat végén pont, felkiáltójel vagy kérdőjel van. A Logo mondatfogalma alkalmas arra, hogy abba több magyar mondatot is beletegyünk pl. [Első mondat. Második mondat!]

Készíts Logo függvényt (darab :mondat), amely megadja, hogy a paraméterében hány magyar mondat van! Ezután add meg a kijelentő, a kérdő és a felkiáltó (felszólító, óhajtó) mondatokat!

darab [Egy. Kettő? Három? Ez a negyedik!] → [4 1 2 1]

```

eljárás darab :mondat
  eredmény számol szám :mondat ". szám :mondat "? szám :mondat "!"
vége

```

Az összes mondat, a kijelentő, kérdő és felkiáltó mondatok számát összefűzi egyetlen listává.

```

eljárás számol :a :b :c
  eredmény (lista :a+:b+:c :a :b :c)
vége

```

A megszámlolja a :jel végű mondatokat.

```

eljárás szám :mondat :jel
  ha üres? :mondat [eredmény 0]
  ha :jel=utolsó első :mondat
    [eredmény 1+szám elsőnélküli :mondat :jel]
  eredmény szám elsőnélküli :mondat :jel
vége

```

## Szólánc

A szólánc játékban a játékosok felváltva mondanak szavakat úgy, hogy a következő szó mindig az előző szó utolsó betűjével kezdődjön. Írj szólánc :a eljárást, amely az :a listában lévő szavakból megállapítja, hogy hány helyes szóláncot tartalmaz, valamint hány szólánc szakadt meg! Egy szólánc helyes, ha legalább 2 szóból áll, az első szavának első betűje azonos az utolsó szavának utolsó betűjével, és nem bontható helyes szóláncokra.

szólánc felbontás [apa avas sas szoba vas sőt tud zug gól] 1 3

Magyarázat:

apa | avas sas szoba | vas sőt tud | zug gól

helyes szólánc: avas sas szoba

megszakadó szólánc: apa, vas sőt tud; illetve zug gól

A bemenő listát át kell alakítanunk listák listájává, amelyekben a szóláncokat helyezzük el! Ezután már egyszerű lesz a dolgunk.

```
eljárás szólánc :a
  szólánc felbontás :a
vége
```

```
eljárás felbontás :a
  eredmény lánccá mindösszevon listává :a elemszám :a 2
vége
```

A listává függvény a szavak listájából, szavakat tartalmazó listákból álló listát készít. Pl. [[vas][lóg][sas]]

```
eljárás listává :a
  ha üres? :a [eredmény []]
  eredmény elsőnek (lista első :a) listává elsőnélküli :a
vége
```

A helyes láncokra bontás azon alapul, hogy először kettő hosszú helyes láncokat képezünk, majd a még fel nem használt szavakból három hosszúakat és így tovább.

```
eljárás mindösszevon :a :n :most
  ha :most>:n [eredmény :a]
  eredmény mindösszevon összevonásokN :a :most :n :most+1
vége
```

N hosszú „helyes” szóláncokat alkot.

```
eljárás összevonásokN :a :n
  ha (üres? :a) [eredmény :a]
  ha (üres? elsőnélküli :a) [eredmény :a]
  ha és (elemszám összevonN :a :n=:n) (helyes? összevonN :a :n)
    [eredmény mondat (lista összevonN :a :n)
      összevonásokN elhagyN :a :n :n]
  eredmény mondat (lista első :a) összevonásokN elsőnélküli :a :n
vége
```

Elhagyja a helyes szóláncot az eredeti listából.

```
eljárás elhagyN :a :n
  ha üres? :a [eredmény :a]
  ha (:n=0) [eredmény :a]
  eredmény elhagyN elsőnélküli :a :n-elemszám első :a
vége
```



Összevonja az :n hosszú helyes láncokat.

```
eljárás összevonN :a :n
  ha vagy (elemszám :a<1) (:n<=0) [eredmény []]
  ha helyes? első :a [eredmény []]
  eredmény mondat első :a összevonN elsőnélküli :a :n-elemszám első :a
vége
```

Listáról eldönti, hogy helyes-e. (Legalább két szóból áll, melynek első és utolsó betűje ugyanaz.)

```
eljárás helyes? :a
  ha elemszám :a<2 [eredmény "hamis]
  ha nem lánc? :a [eredmény "hamis]
  eredmény első első :a=utolsó utolsó :a
vége
```

Eldönti egy listáról, hogy lánc-e.

```
eljárás lánc? :a
  ha üres? :a [eredmény "igaz]
  ha üres? elsőnélküli :a [eredmény "igaz]
  ha első első elsőnélküli :a <> utolsó első :a [eredmény "hamis]
  eredmény lánc? elsőnélküli :a
vége
```

A különálló szavakból és helyes szóláncokból álló listában összefűzi a megszakadó láncokat.

```
eljárás lánccá :a
  ha üres? :a [eredmény []]
  ha üres? elsőnélküli :a [eredmény :a]
  ha lánccá? első :a első elsőnélküli :a
    [eredmény lánccá elsőnek mondat első :a első elsőnélküli :a
    elsőnélküli elsőnélküli :a]
  eredmény elsőnek első :a lánccá elsőnélküli :a
vége
```

Eldönti, hogy lánccá összefűzhető-e két szólánc. Akkor fűzhető össze, ha nem „helyes” lánc és az első lánc utolsó betűje azonos a második szólánc első betűjével.

```
eljárás lánccá? :l1 :l2
  eredmény és (utolsó utolsó :l1=első első :l2)
    (nem vagy helyes? :l1 helyes? :l2 )
vége
```

Végiglépked az átalakított listán, amelyben már egy-egy listává fűztük a megszakadó és helyes listákat és megszámlálja ezeket.

```
eljárás szólánc1 :a
  ha üres? :a [eredmény [0 0]]
  ha helyes? első :a [eredmény elsohoz szólánc1 elsőnélküli :a]
  eredmény másodikhoz szólánc1 elsőnélküli :a
vége
```

Két elemből álló lista első eleméhez egyet ad, vagyis számolja a helyes láncokat.

```
eljárás elsohoz :a
  eredmény lista 1+első :a utolsó :a
vége
```

Két elemből álló lista második eleméhez ad hozzá egyet, vagyis számolja a megszakadó láncokat.

```
eljárás másodikhoz :a
  eredmény lista első :a 1+utolsó :a
vége
```

## Irodalomjegyzék

- Abonyi-Tóth Andor, Holler János, Rozgonyi-Borus Ferenc: Képzeld el! Imagine - technócgrafika, multimédia és játékok, ABAX Kiadó, 2007
- Abonyi-Tóth Andor, Holler János, Rozgonyi-Borus Ferenc: Képzeld el! Imagine - algoritmusok, játékok, ABAX Kiadó, 2008
- Logo versenyfeladatok tára, 1998-2002, NJSzT, Budapest, 2013
- Logo versenyfeladatok tára, 2003-2007, NJSzT, Budapest, 2013
- Logo versenyfeladatok tára, 2008-2012, NJSzT, Budapest, 2013
- A Logo OSzTV honlapja: <http://logo.inf.elte.hu/>

<b>Logo versenyfeladatok</b> .....	0
Bevezető .....	1
Sokszögek, csillagok .....	2
Derékszögű háromszögekből álló alakzatok .....	2
Elemi feladatvariációk .....	2
Szabályos háromszöget tartalmazó alakzatok .....	3
Háromszögek illesztése szabályos sokszögek oldalára .....	3
Háromszögek illesztése szabályos sokszögek oldalára, a körvonal megrajzolásával .....	3
Variációk .....	4
Szabályos háromszög illesztése szabályos sokszögek csúcsaira .....	4
Háromszögek szembefordítása, masnik rajzolása .....	6
Szabályos háromszögekből álló jelvények rajzolása .....	7
1. variáció .....	7
2. variáció .....	8
Négyszögek rajzolása .....	9
Négyzetből alkotott alakzatok készítése .....	9
Trapézból alkotott alakzatok rajzolása .....	10
Hatszögvariációk .....	10
1. feladat .....	10
2. feladat .....	11
Nyolcszögek rajzolása .....	12
Törött sokszög rajzolása .....	13
Csillagok rajzolása .....	13
Szabályos csillagsokszögek .....	13
Csillagokból álló komplex ábrák .....	17
<b>Körök, körívek rajzolása</b> .....	18
Körvonal rajzolása (ha a teknőc pozíciója a körvonalon van) .....	18
Elemi feladatvariációk .....	20
Kitöltött kör rajzolása (ha a teknőc pozíciója a körvonalon van) .....	21
Elemi feladatvariációk .....	21
Kör rajzolása (ha a teknőc pozíciója a kör közepén van) .....	22
Elemi feladatvariációk .....	23
Kitöltött kör rajzolása (ha a teknőc pozíciója a kör közepén van) .....	24
Elemi feladatvariációk .....	24
Félkör rajzolása .....	24

---

Elemi feladatvariációk .....	26
Körívek rajzolása.....	26
Feladatvariációk.....	28
<b>Variációk zászlók rajzolására .....</b>	<b>35</b>
1. feladatsor: az alapmegoldás programjából kiindulva .....	36
Elemi feladatvariációk: .....	37
2. feladatsor: Vízszintes és függőleges csíkok együttes alkalmazása .....	39
Elemi feladatvariációk: .....	39
3. feladatsor: Keresztek alkalmazása.....	42
Elemi feladatvariációk: .....	44
4. feladatsor: Csíkok és csillagok együttes alkalmazása .....	46
Elemi feladatvariációk: .....	47
5. feladatsor Körök megjelenése a zászlókon .....	51
Elemi feladatvariációk: .....	52
6. feladatsor: Háromszögek .....	54
Elemi feladatvariációk: .....	54
<b>Térkitöltés forgatással.....</b>	<b>57</b>
Forgatott elemek .....	57
Jégvirág .....	57
Virágablak.....	57
Jégvirág – 1. változat.....	58
Jégvirág – 2. változat.....	59
Virágszirmok.....	60
Rombuszok .....	61
Penrose-mozaik – 1. változat .....	62
Penrose-mozaik – 2. változat .....	63
Penrose-mozaik – 3. változat .....	64
Hatszögek.....	65
Háromszögek.....	66
Virág.....	67
Mozaik .....	68
Forgó.....	68
Díszüveg.....	69
<b>Sorminták.....</b>	<b>71</b>
Sorminta azonos elemekből.....	71

---

---

Írásos hímzés .....	71
Fűrész.....	72
Nyomok.....	72
Sokszögek.....	73
Sorminta egy alapelemmel és más záró elemmel.....	74
Nyaklánc.....	74
Égig érő paszuly .....	75
Autóbusz .....	75
Sorminta elforgatott alapelemmel.....	76
Sorminta .....	76
Sorminta két alapelemből.....	77
Hullámok.....	77
Lépcső.....	78
Zongora.....	78
Sorminta több alapelemből, záró elemmel.....	79
Nyaklánc.....	79
Sorminta sorból és tükörképéből összeállítva.....	81
Kígyó-sárkány .....	81
Írásos hímzés, 2. verzió .....	82
Írásos hímzés, 3. verzió .....	82
Sorminta több sorból.....	84
Négyzetek.....	84
Tányértartó.....	85
Fríz .....	86
Szálbehúzásos hímzés.....	86
Szálbehúzásos hímzés, 2. verzió .....	87
Szálbehúzásos hímzés, 3. verzió .....	89
Szálbehúzásos hímzés, 4. verzió .....	90
<b>Mozaik – sorminták egymás fölé .....</b>	<b>92</b>
Négyzetmozaikok azonos alapelemekből.....	92
Csempe .....	92
Csempe kihagyott területtel .....	93
Csempe külső szegély felhasználásával .....	94
Csempe látszólag szembefordított sorokkal.....	94
Asztal – két színnel festve, az egyik a háttér .....	96

---

---

Padló – két szín, újrafestéssel .....	97
Terítő – két mozaik egymáson .....	98
Csillagok .....	99
Mozaik kétféle sorból .....	99
Csempe .....	99
Ablak – a külső elemek más színűek .....	101
Mozaik kétféle elemből.....	102
Padló .....	102
Csillagok .....	103
Mozaik – szegély is, kétféle elem is.....	104
Nem négyzet alapú mozaikok .....	106
Háló.....	106
Hatszögmozaik .....	107
Síklefedés körökkel .....	108
<b>Ásványok – molekulák – kristályok .....</b>	<b>110</b>
Sokszögrajzolásra épülő ásványok .....	110
Gyémántok.....	110
Szilikát.....	111
Gráfrajzolásra épülő ásványok .....	112
Szilikát – első változat .....	112
Szilikát – második változat.....	113
Szilikát – harmadik változat.....	113
Piroxén.....	114
Amfiból.....	115
Csillám – első változat.....	116
Csillám – második változat .....	117
Béta-kvarc.....	118
Sassolin .....	120
<b>Rekurzió .....</b>	<b>122</b>
Sorminta változó méretű elemekkel .....	122
Csökkenő háromszögek .....	122
Távolodó madarak .....	122
Jégcsap .....	123
Vágányok.....	124
Orgonasíp.....	124

---

Spirálok .....	125
Tekergő.....	125
Spirál négyzetekből.....	126
Színes spirál.....	126
Kettős spirál.....	127
Kettős spirál háromszögekből.....	128
Spirál betűkből.....	129
Csigavonal .....	130
Spirál-sokszög.....	130
Rekurzív forgatás, eltolás .....	131
Körcikkekből spirál.....	131
Legyező.....	131
Pálcikák.....	132
Íves elemek.....	133
<b>Mozaik – rekurzívan.....</b>	<b>134</b>
Forgatott elemek .....	134
Mozaik 90 fokkal forgatott alapelemekkel.....	134
Mozaik forgatott alapelemekkel.....	135
Mozaik többféle elemmel.....	136
Kockás abrosz – 1. változat.....	136
Kockás abrosz – 2. változat.....	137
Mozaik kétféle elemből közvetett rekurzióval.....	138
Különböző méretű elemek .....	139
Maja templom.....	139
Színház.....	140
Sorok hossza különböző .....	141
Piramis .....	141
Hatszögmozaikok.....	141
<b>Optikai csalódások .....</b>	<b>143</b>
Néhány klasszikus példa.....	143
Térbeli (paradox) illúziók.....	143
Penrose háromszög.....	143
Térhatás színekkel.....	144
Torzító illúziók .....	144
Müller-Lyer-illúzió .....	144

---

---

Kávéház fal illúzió.....	145
Görbe oldalú négyzet.....	146
Ferde csíkok.....	147
Fekete-fehér illúzió.....	149
Nyilak.....	149
Körök és háromszögek .....	150
Hullámváz.....	150
Körök vagy négyzetek?.....	152
Eltűnő és előtűnő elemek.....	154
Vibráló pontok .....	154
Mozgás látszata.....	154
Forgó mandulák .....	155
Forgó körök.....	156
<b>Variációk fa rajzolásra.....</b>	<b>157</b>
1. Feladatsor az alapmegoldás programjából kiindulva .....	157
Elemi feladatvariációk: .....	158
2. Feladatsor az alapmegoldás eredményéből kiindulva: .....	161
Ágak a törzsből.....	161
Az ágak egy része nem fa .....	162
Ágak oldalai külön rajzolva.....	163
3. Feladatsor két megoldás kombinálásával:.....	164
4. Feladatsor külső hatásokkal befolyásolva: .....	167
Paraméterfüggő rajzok .....	167
Írányfüggő rajzok .....	169
Véletlenszerű fák.....	171
5. Fák más alapelemekből .....	171
<b>Fraktálok .....</b>	<b>174</b>
Szakasz helyettesítése egy törött vonallal.....	175
Sierpinski nyílhegy görbéje.....	175
Ugyanarra a szakaszra két, a szakaszra tükrözött töröttvonalat illesztünk .....	176
Kiindulás nem egyetlen szakaszból, egy háromszög oldalaiból.....	176
Nem minden szakaszra illesztünk töröttvonalat.....	177
Paraméterrel megadott számú szakaszból indulunk ki .....	177
Háromszögek felhasználása fraktál alakzatok készítésénél .....	178
Az oldalak közepén újabb háromszögek nőnek ki .....	178



A háromszög közepéből kivágunk háromszögeket.....	178
Új háromszögek a csúcsonál .....	179
Hasonló háromszögekre darabolt háromszög(ek) .....	180
Fraktálok négyzetekből építve .....	181
Sierpinski négyzet, a középső négyzetet kivágjuk.....	181
Darabolás négyzetekkel.....	182
Darabolás, csücskök levágása .....	182
Darabolás téglalapokkal.....	183
Fraktálok körökből építve.....	184
Körvonalon elhelyezkedő újabb körök.....	184
Nem minden fraktálszint látszódik.....	184
Körben körívek – Jinjang.....	185
Fraktálkészítés többféle sokszög felhasználásával.....	186
A sokszög sarkaiban kifelé rajzolva újabb sokszögek.....	186
A sokszög sarkaiban befelé rajzolva újabb sokszögek.....	186
Vezérlő eljárással készülő fraktálok .....	186
Jégvirág .....	186
Növény .....	187
Bimbó .....	188
Peano görbe .....	189
Szintenként másként viselkedő fraktálok.....	190
Trapézok, páros-páratlan vizsgálattal .....	190
Hatszög, egymást hívó rekurzív eljárásokkal.....	190
Növény, újabb fraktálok megjelenése .....	191
Csipke .....	193
Rajzok, mint a fraktálrajz elemei.....	193
Halak, 1. verzió.....	193
Halak, 2. verzió.....	194
Halak, 3. verzió.....	195
Számításokkal vezérelt rajzolás.....	197
Kezdőpont transzformációja.....	197
Hőmérő .....	197
Elfordulás szögének kiszámítása.....	198
Óra .....	198
Óra, 2. variáció .....	199

Térfogatszámítás.....	200
Homokóra.....	200
Kád.....	200
Üdítő.....	201
Tejeskávé.....	202
Számrendszerek közötti átváltás.....	203
Bináris szám kirajzolása.....	203
Vegyes alapú számrendszerek közötti átváltás.....	204
Címletező.....	204
Súlyzó.....	205
Római számok.....	206
A környezet érzékelésével vezérelt rajzolás.....	207
Rajzolás vonalig.....	207
Átlós sokszög.....	207
Csíkos kör.....	207
Hatszög.....	208
Ívek.....	209
Irányfüggő rajzok.....	210
Irányfüggő fa.....	210
Szöveggel, listákkal vezérelt rajzolás.....	211
Interaktív vezérlés karakterekkel.....	211
Tanulás.....	211
Vezérlés karaktersorozattal.....	212
Meander, fordulások vezérlése.....	212
Ovilogo, a fordulások és a lépések nagyságának vezérlése.....	213
Görögös sorminta, színezés vezérlése.....	214
Képrajzoló, pozíció vezérlése.....	215
Morze, a rajzolás alakjának vezérlése.....	215
Rovásírás.....	216
Titkosírás, alak és színek vezérlése.....	218
Vezérlés számok listájával.....	219
Minta, alak vezérlése.....	219
Időjárás, méret vezérlése.....	219
Időjárás, 2. verzió.....	220
Képrajzoló, pozíció vezérlése.....	220

Vezérlés szavak listájával.....	221
Nóta, pozíció vezérlése újabb lista beiktatásával.....	221
Vezérlés vegyes listával.....	223
Kotta, alak és pozíció vezérlése.....	223
Parkolás, színek és pozíció vezérlése.....	224
Vezérlés két listával.....	225
Parkolás, 2. verzió.....	225
Vezérlés listák listájával.....	227
Rovásírás, alak kirajzolása.....	227
Titkosírás, pozíció vezérlése.....	229
Mozaik.....	230
<b>Szövegmanipuláció.....</b>	<b>231</b>
A feladat eredménye lista.....	231
Bumm játék.....	231
A feladat bemenete és eredménye is szó.....	232
Titkosírás.....	232
Akasztófa játék.....	232
A feladatnak két szó a bemenete.....	233
Anagramma.....	233
A feladatnak két lista a bemenete.....	233
Kártyázás.....	233
Kártya.....	234
A feladat bemenete és eredménye is mondat/lista.....	235
Ékezetek.....	235
Számok.....	235
Ezresek.....	236
Római számok.....	237
Mondatok.....	238
Szólánc.....	239
<b>Irodalomjegyzék.....</b>	<b>241</b>