

## VASTAG PRIMITÍVEK RAJZOLÁSA

Képpontok ismétlése  
Mozgó ecset  
Területkitöltés  
Közelítés vastag szakaszokkal

## VASTAG PRIMITÍVEK RAJZOLÁSA

Több képpontnyi vastagságú vonalak

Milyen alakú legyen az ecset?

Kör?

Téglalap?

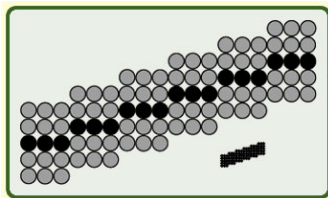
Forduljon a vonallal?

## VASTAG PRIMITÍVEK RAJZOLÁSA

### 1. Képpontok ismétlése

A pásztázó vonalas algoritmus kiterjesztése:

ha  $-1 < m < 1$ , akkor a képpontokat többszörözzük meg az oszlopokban; különben a sorokban

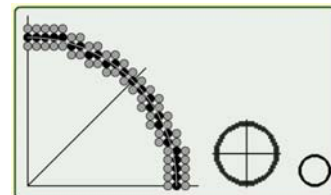


## Képpontok ismétlése

Tulajdonságai:

- a) gyors,
- b) a vonal végei mindig vízszintesek vagy függőlegesek,
- c) a vonal vastagsága függ a meredekségtől

- d) a duplázás nem megy: a vonal valamelyik oldala felé vastagabb

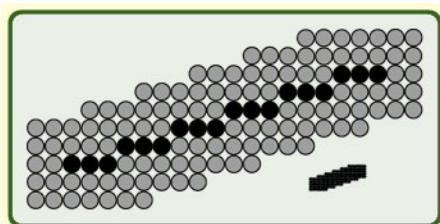


Jó módszer, ha nem túl vastag a vonal

## VASTAG PRIMITÍVEK RAJZOLÁSA

### 2. Mozgó ecset

Téglalap alakú „ecset”, aminek a középpontja (vagy csúcspontja) az 1 pixel vastag vonalon mozog (az ecset nem "forog")

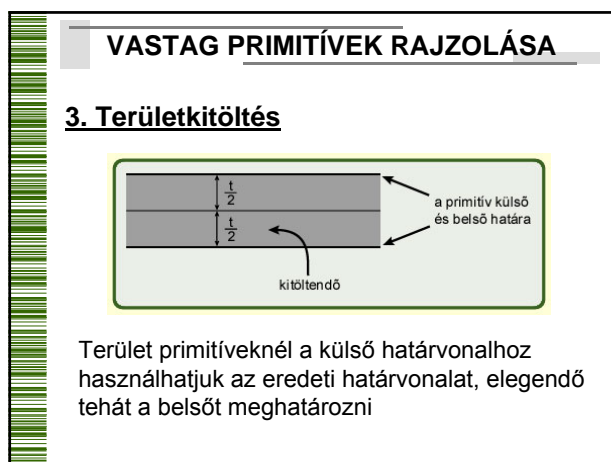
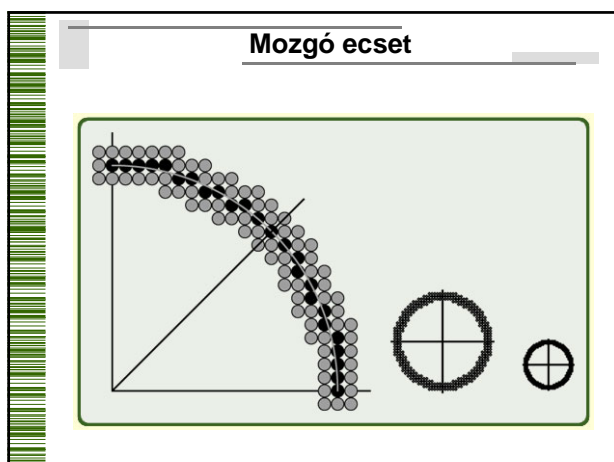


## Mozgó ecset

Tulajdonságai:

- hasonló 1-hez, de
- a) a végpontok „nagyobbak”
- b) a vonal vastagsága függ a meredekségtől és az ecset alakjától
- jobb a kör alakú ecset

**Implementáció:** ecset (= minta) másolása az 1 pixel vastag vonal minden pontjába



### Területkitöltés

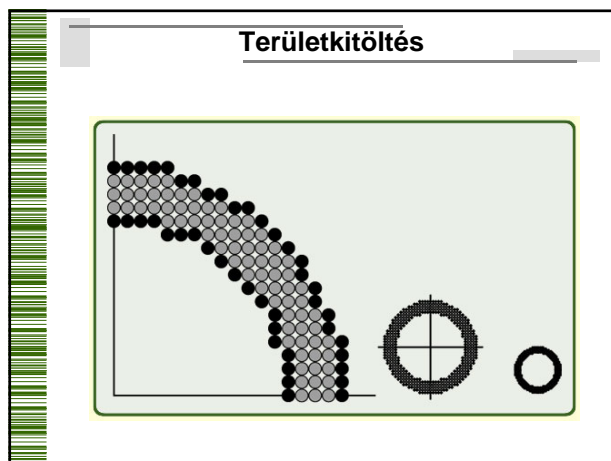
**Tulajdonságai:**

- a) ugyanolyan jó páros és páratlan vastagra
- b) a vonal vastagsága nem függ a meredekségtől

Kör esetén: külső és belső kör

Ellipszis esetén:

$a - t/2, b - t/2$	belső	} ellipszisek
$a + t/2, b + t/2$	külső	



### VASTAG PRIMITÍVEK RAJZOLÁSA

#### 4. Közelítés vastag szakaszokkal

Szakaszonként lineáris approximáció

- a) szép
- b) vastag vonalakat simán kell illeszteni

### Pont mérete (OpenGL)

```

Void glPointSize(GLfloat size);

Nem minden méretet támogatnak az
implementációk:

GLfloat sizes[2]; // méret tartomány
GLfloat step; // támogatott lépés

glGetFlov(GL_POINT_SIZE_RANGE,
sizes);

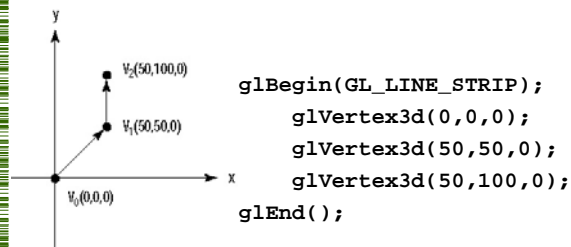
glGetFlov(GL_POINT_SIZE GRANULARITY

```

### Szakaszok rajzolása (OpenGL)

#### Szakasz sorozat (GL\_LINE\_STRIP):

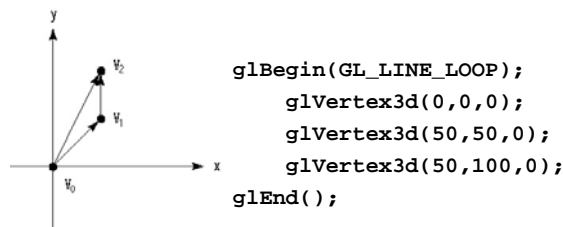
Egy vagy több összekötött szakasz specifikálása a végpontok sorozatának megadásával. Pl.:



### Szakaszok rajzolása (OpenGL)

#### Szakasz hurok (GL\_LINE\_LOOP):

Ugyanaz, mint a LINE\_STRIP, de az utolsóként specifikált csúcspontot összekötjük az elsőként specifikált csúcsponttal. Pl.:



### Szakaszok rajzolása (OpenGL)

#### Független szakasz (GL\_LINES):

Az elsőként specifikált két csúcspont határozza meg az első szakaszt, a második két csúcspont a második szakaszt, ... (nincsenek összekötve)

### Vonal vastagsága (OpenGL)

```
Void glLineWidth(GLfloat width);
```

Nem minden vastagságot támogatnak az implementációk:

```
GLfloat sizes[2]; // vastagság tartomány
GLfloat step; // támogatott lépés
```

```
glGetFlov(GL_LINE_WIDTH_RANGE, sizes);
glGetFlov(GL_POINT_SIZE_GRANULARITY,
&step);
```

### Szakaszok élsimítása (OpenGL)

A szakaszok élsimítását engedélyezni a GL\_LINE\_SMOOTH argumentummal meghívott glEnable, tiltani a glDisable függvénnyel lehet.

Ha az élsimítás engedélyezett, akkor nem egész szélességek is megadhatók, és ekkor a szakasz szélén kirajzolt képpontok intenzitása kisebb, mint a szakasz közepén lévő képpontoké.