

# **Matematická morfológia**

Cvičenia z Počítačového Videnia

# Čo už vieme

Načítať obrázok

Konvertovať na šedoúrovňový

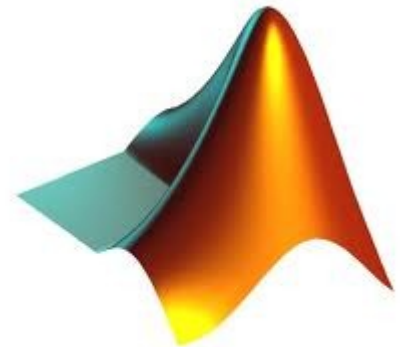
Ekvalizovať histogram

Vyhladiť

Prahovať

Nájsť hrany

<= Binárny obrázok

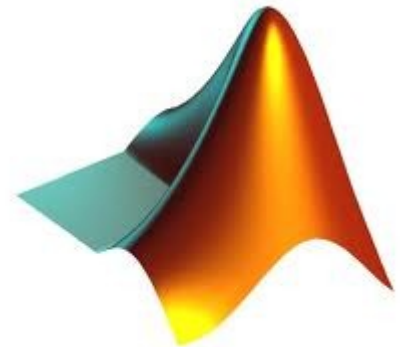


# Čo ďalej

## Matematická morfológia

### Hľadanie objektov:

- Houghova transformácia
- Počet objektov
- Veľkosť, tvar...



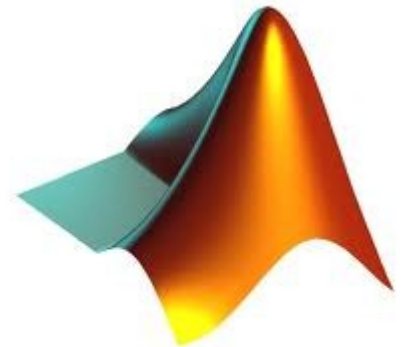
# Doplnenie k minulotýždňovým cvičeniam

## Separabilné filtre

Konvolúcia maskou  $k \times k$  stojí  $k^2$  operácií pre každý pixel

Niektoré masky môžu byť rozdelené na 1D masky – horizontálne a vertikálne

Cena sa zníži na  $2k$  operácií



# Separabilné filtre

$$\frac{1}{K^2} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & 1 & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

$$\frac{1}{8} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

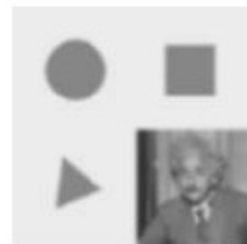
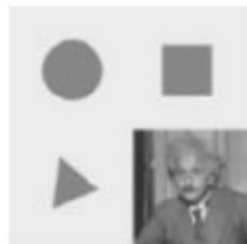
$$\frac{1}{4} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

$$\frac{1}{K} \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}$$

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

$$\frac{1}{16} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

$$\frac{1}{2} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$$

$$\frac{1}{2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$$


(a) box,  $K = 5$

(b) bilinear

(c) "Gaussian"

(d) Sobel

(e) corner

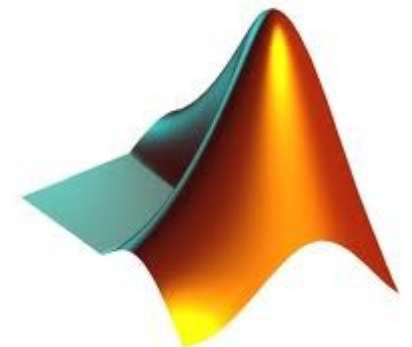
# Matematická morfológia

- pracuje s transformáciami, ktoré sú opísané s použitím operátorov v nelineárnej algebre
- predstavuje protipól ku tradičnému spracovaniu signálov, ktoré používa lineárne operátory (napr. konvolúcia)

## Morfológia

binárna (2D bodová množina)

šedotónová (3D bodová množina)

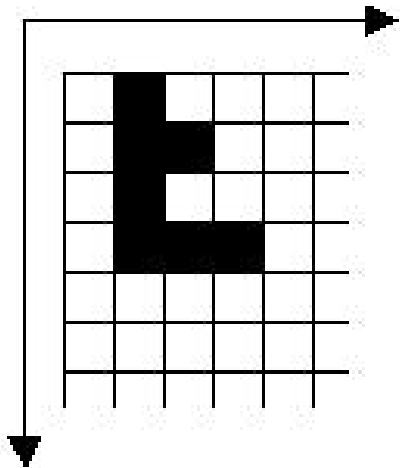


# Matematická morfológia I.

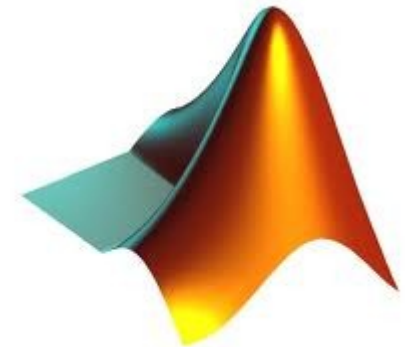
- predpokladá, že obraz sa dá modelovať pomocou bodových množín

## Bodová množina

- množina súradníc obrazových bodov, ktoré patria objektu

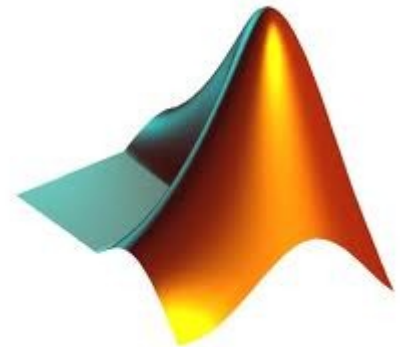


$$X = \{(1, 0), (1, 1), (1, 2), (1, 3), (2, 1), (2, 3), (3, 3)\}$$



# Použitie

- predspracovanie
  - odstránenie šumu, zjednodušenie tvaru
- tvorba kostry, stenčovanie/zhrubnutie obrazu, tvorba konvexného obalu...
- segmentácia



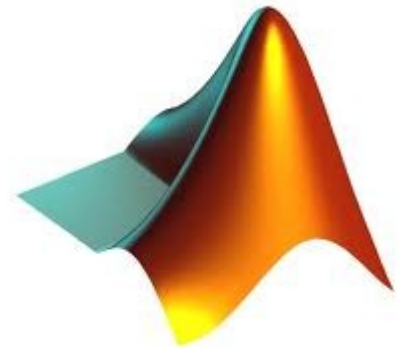
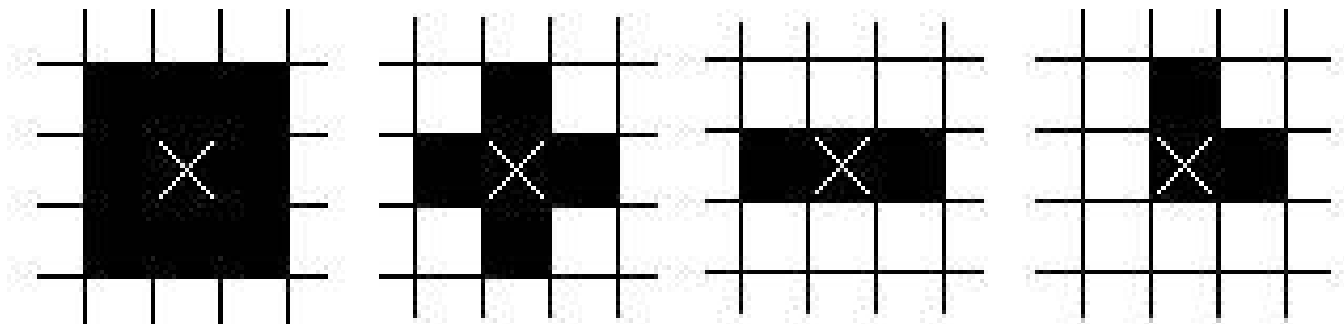


# Morfologická transformácia

- relácia medzi bodovou množinou  $X$  a štruktúrálnym elementom  $B$

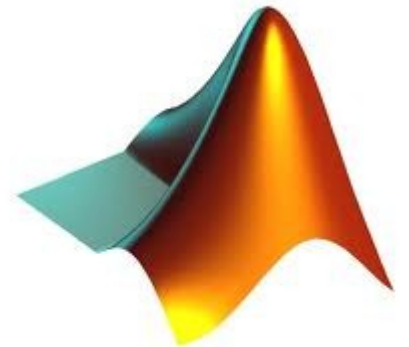
## Štruktúrálny element:

- bodová množina
- obsahuje jeden reprezentatívny bod  $O$



# Morfologické transformácie

- dilatácia, erózia
- opening (otvorenie), closing (uzavretie)
- hit-or-miss
- translácia, reflexia, komplement, rozdiel, extrancia hranice, nájdenie konvexného obalu, stenčenie, zhrubnutie, nájdenie kostry...

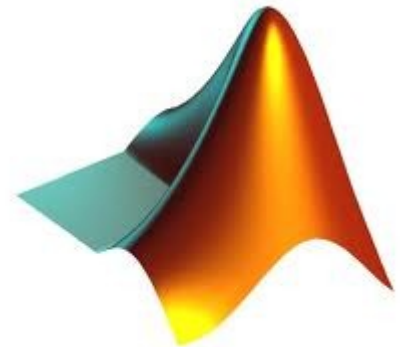


# Dilatácia

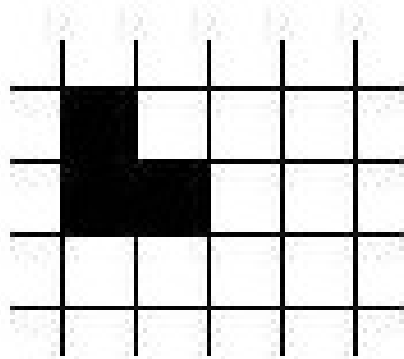
- založená na Minkowského súčte

$$X \oplus B = \{p \in \varepsilon^2, p = x + b, x \in X, b \in B\}$$

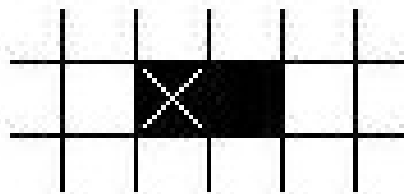
- relácia, ktorá skladá body dvoch množín pomocou vektorového súčtu
- zjednotenie posunutých bodových množín
- komutatívna, asociatívna, invariantná na otočenie



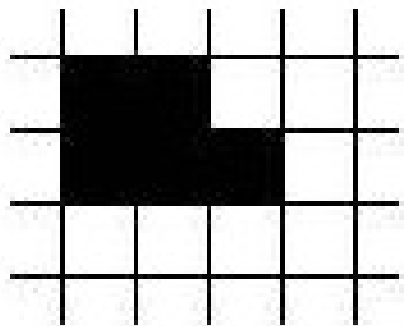
# Dilatácia



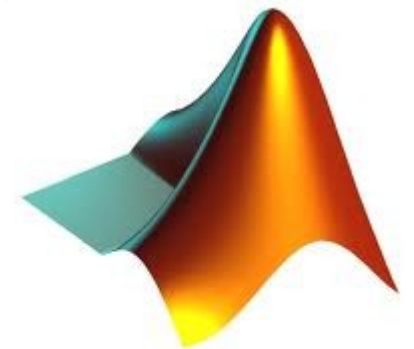
$$X = \{(1, 0), (1, 1), (2, 1)\}$$



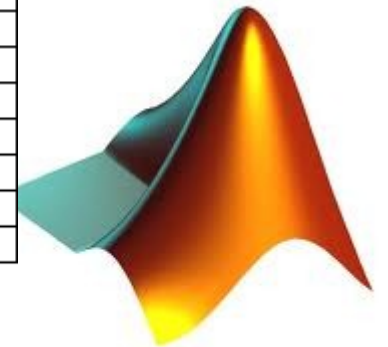
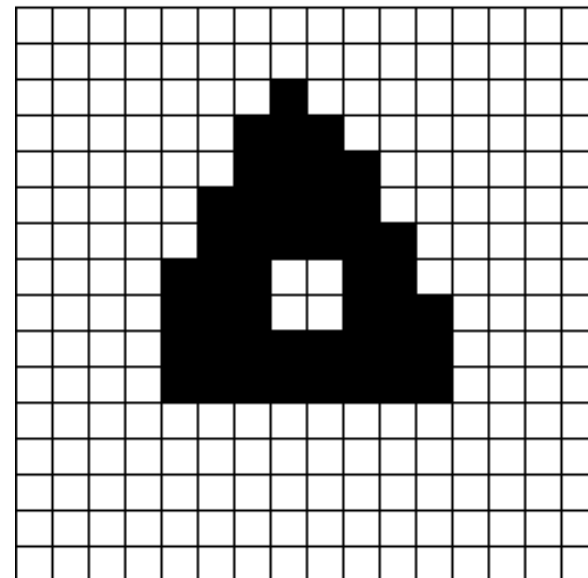
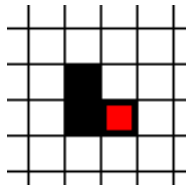
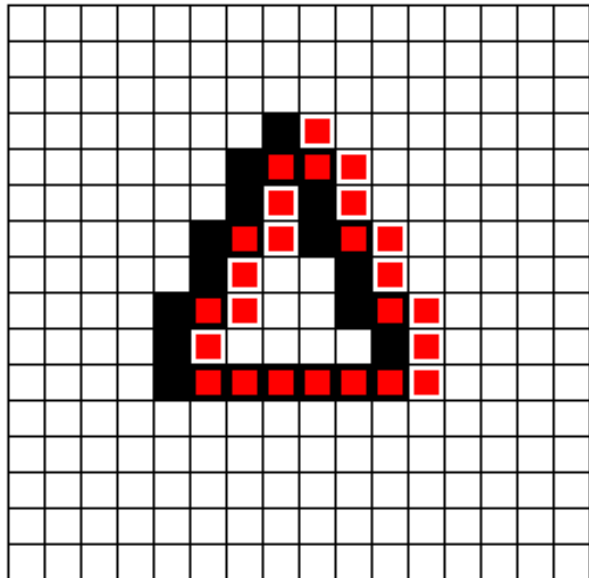
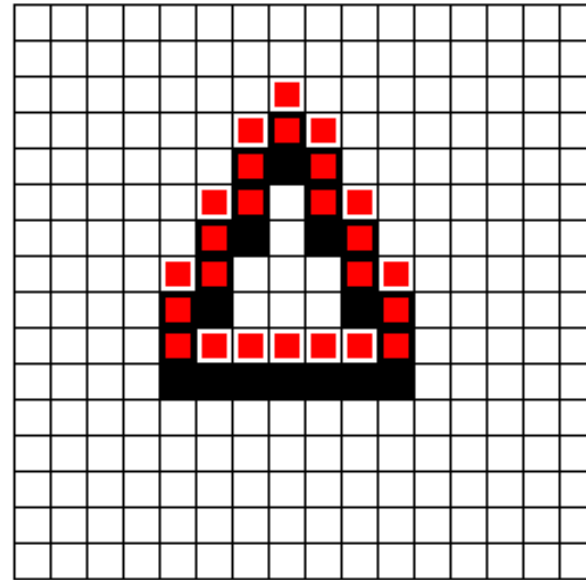
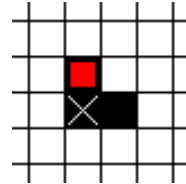
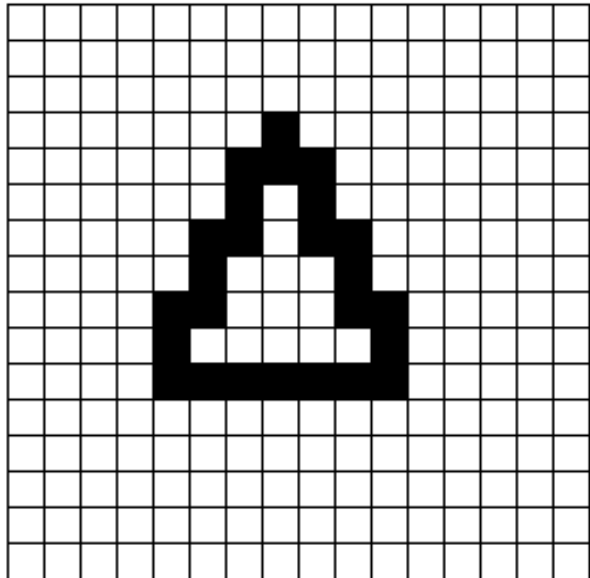
$$B = \{(0, 0), (1, 0)\}$$



$$X \oplus B = \{(1, 0), (1, 1), (2, 1) \\ (2, 0), (2, 1), (3, 1)\}$$

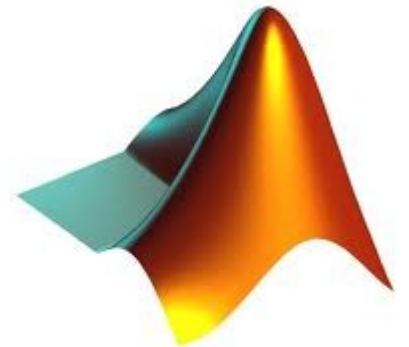


# Dilatácia

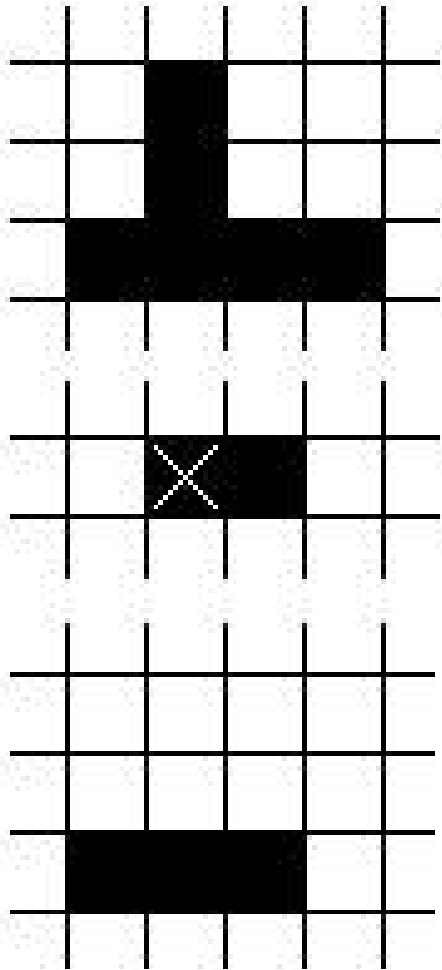


# Erózia

- založená na Minkowského rozdieli  
$$X - B = \{p \in \mathbb{R}^2, p+b \in X, b \in B\}$$
- komutatívna, invariantná voči posunutiu
- prienik všetkých posunutí



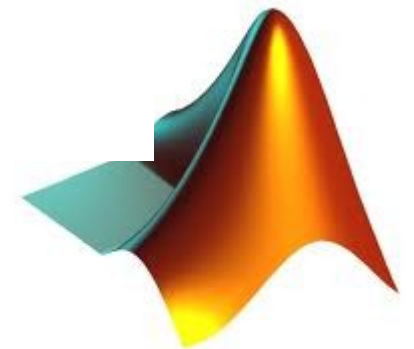
# Erózia



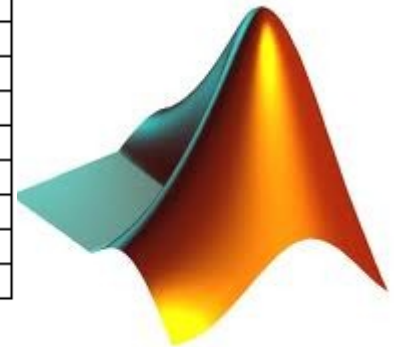
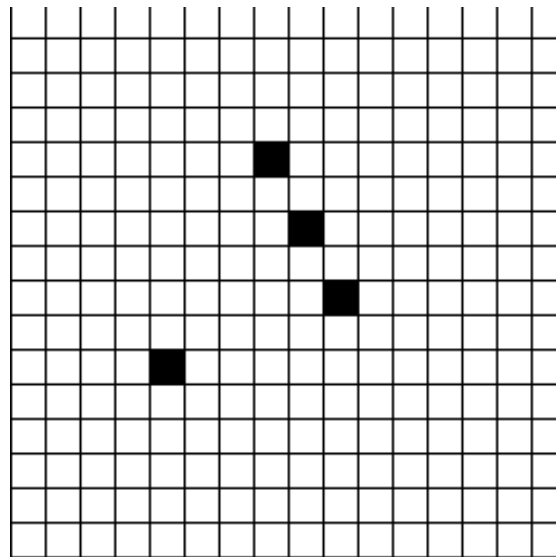
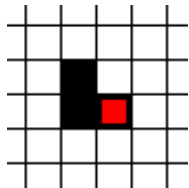
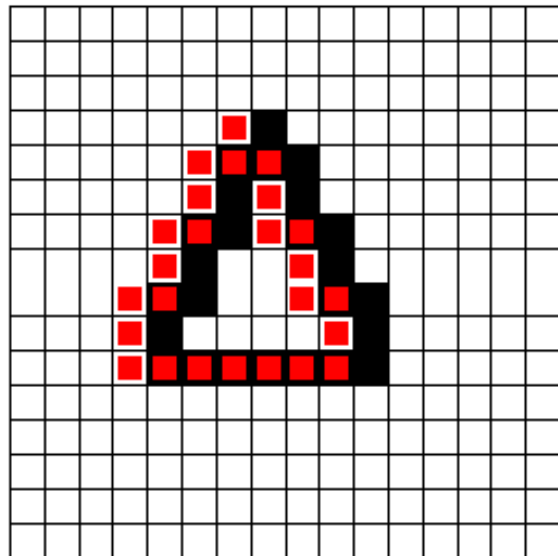
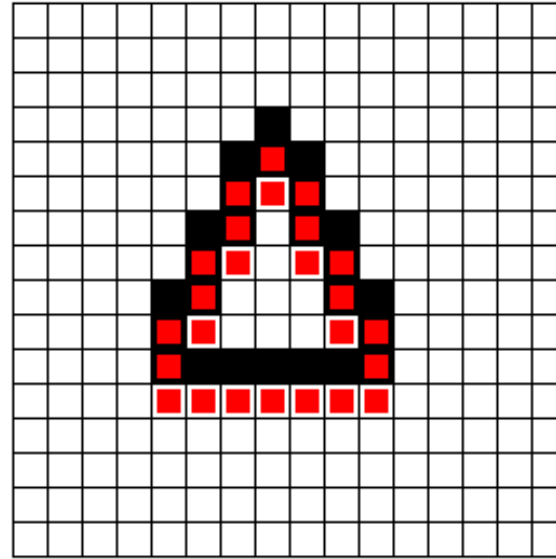
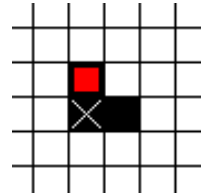
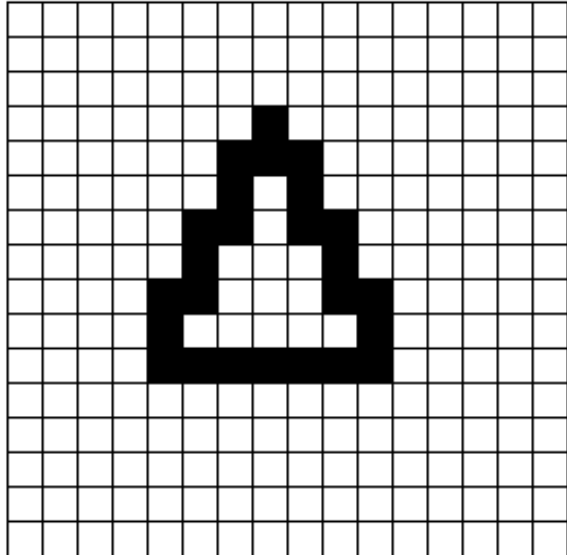
$$X = \{(1, 0), (1, 1), (1, 2), (0, 2), \\ (1, 2), (2, 2), (3, 2)\}$$

$$B = \{(0, 0), (1, 0)\}$$

$$X \ominus B = \{(0, 2), (1, 2), (2, 2)\}$$

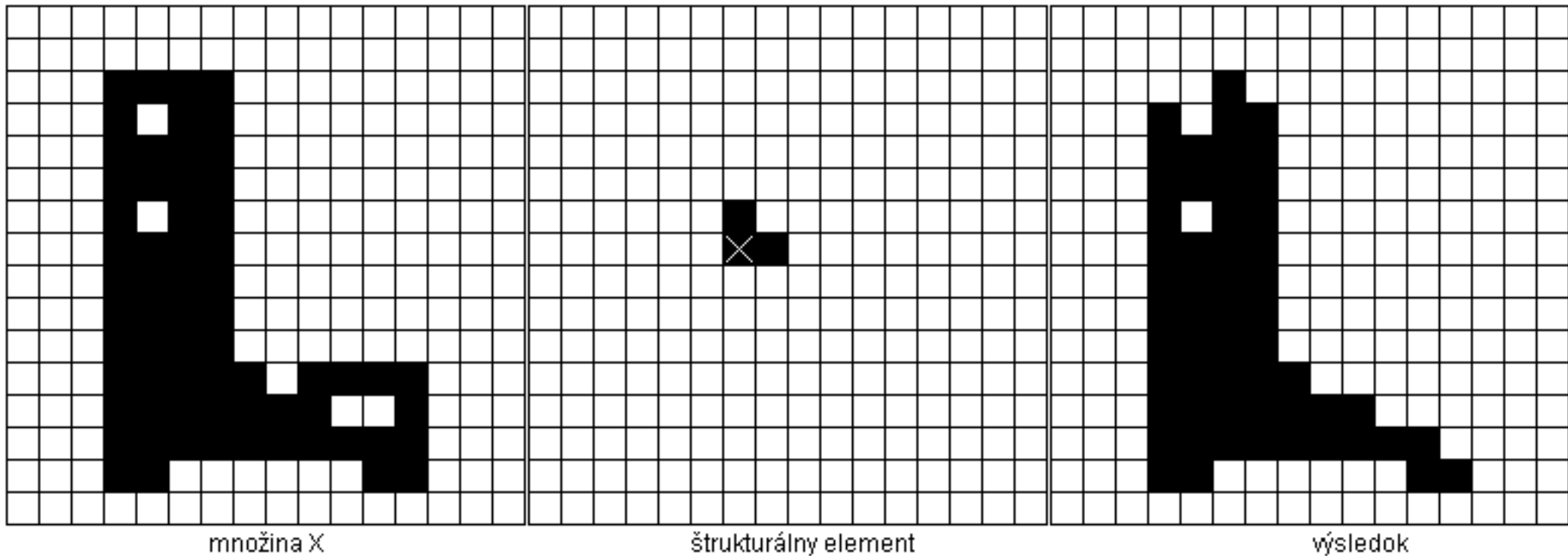


# Erózia

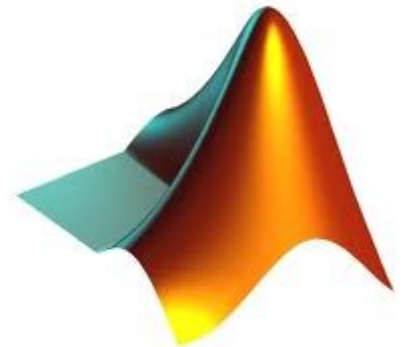




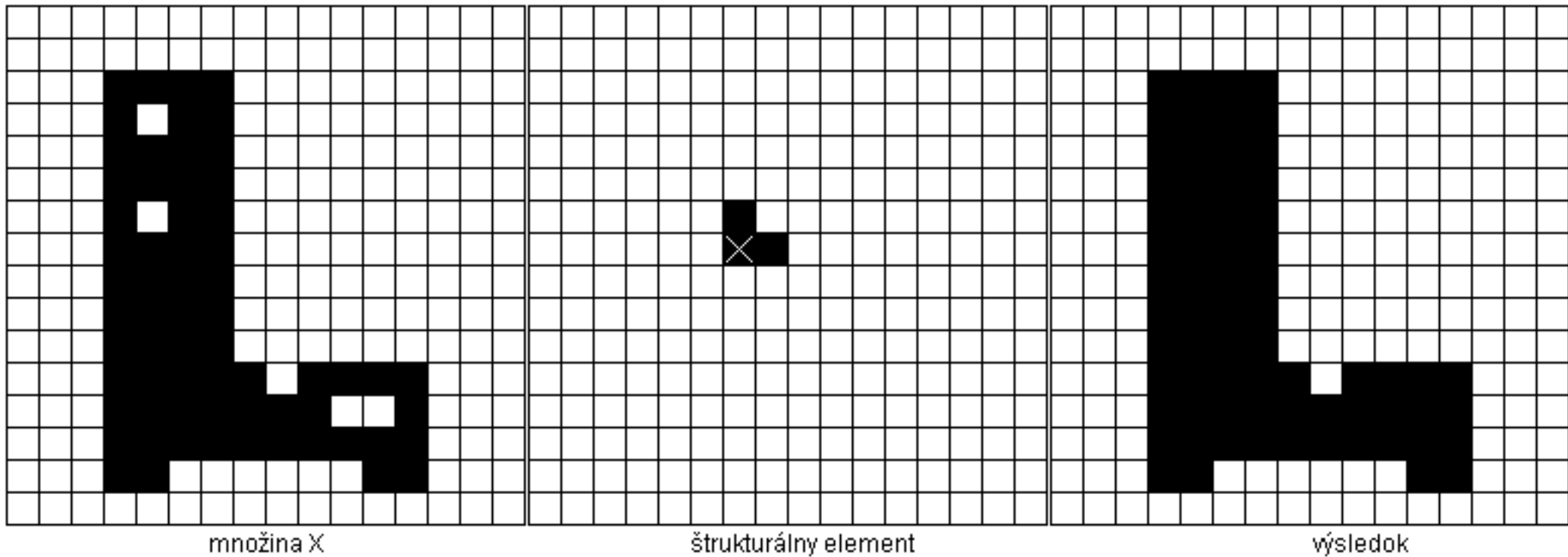
# Opening



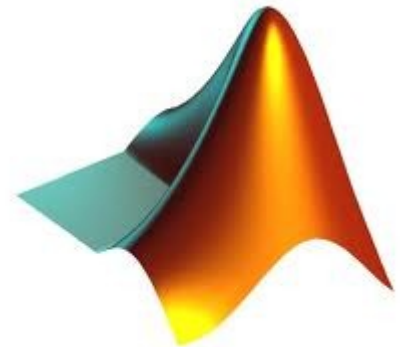
- erózia za ktorou nasleduje dilatácia
- $X \circ B = (X - B) \oplus B$



# Closing

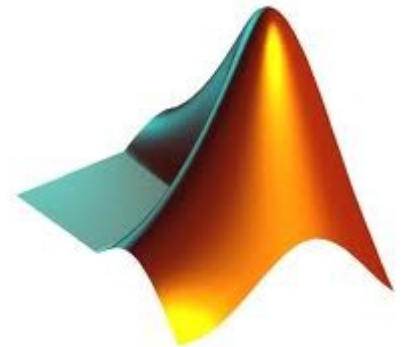


- dilatácia , za ktorou nasleduje erózia
- $X \cdot B = (X \oplus B) - B$



# Opening a Closing

- obe operácie sú idempotentné: ich opätovná aplikácia nemení predošlý výsledok
- použitie rovnakého štrukturálneho prvku
- duálne operácie – zatvorenie popredia dá rovnaký výsledok, ako otvorenie pozadia



# Hit and miss

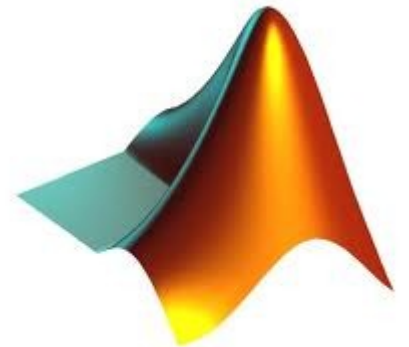
- hľadanie hrán a rohov objektov, nájdenie izolovaných bodov, koncových bodov kostry
- štruktúrálny element
  - 1,0,don't care
  - závislý na tom, čo hľadám
- príklad: nájdenie rohov

	1	
0	1	1
0	0	

	1	
1	1	0
	0	0

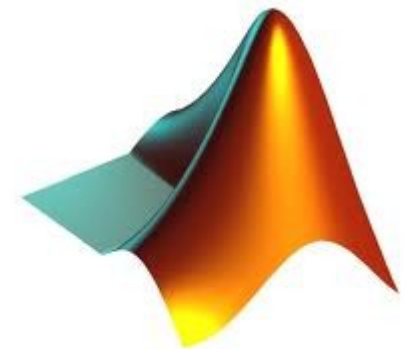
	0	0
1	1	0
	1	

0	0	
0	1	1
	1	



# Iné operácie

- thinning
  - $\text{thin}(X,B) = X - \text{hit\_and\_miss}(X,B)$
- thickening
  - $\text{thicken}(X,B) = X \cup \text{hit\_and\_miss}(X,B)$
- skeletonization
- $X$  – obraz,  $B$  – štrukturálny prvok



# Hľadanie hranice

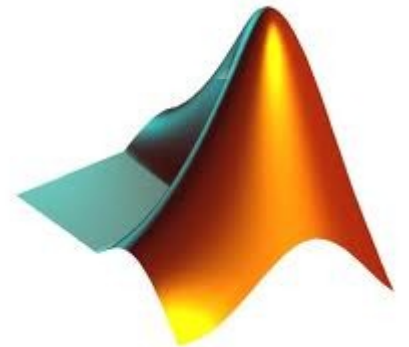
- použitie dilatácie a erózie

- **Algoritmy**

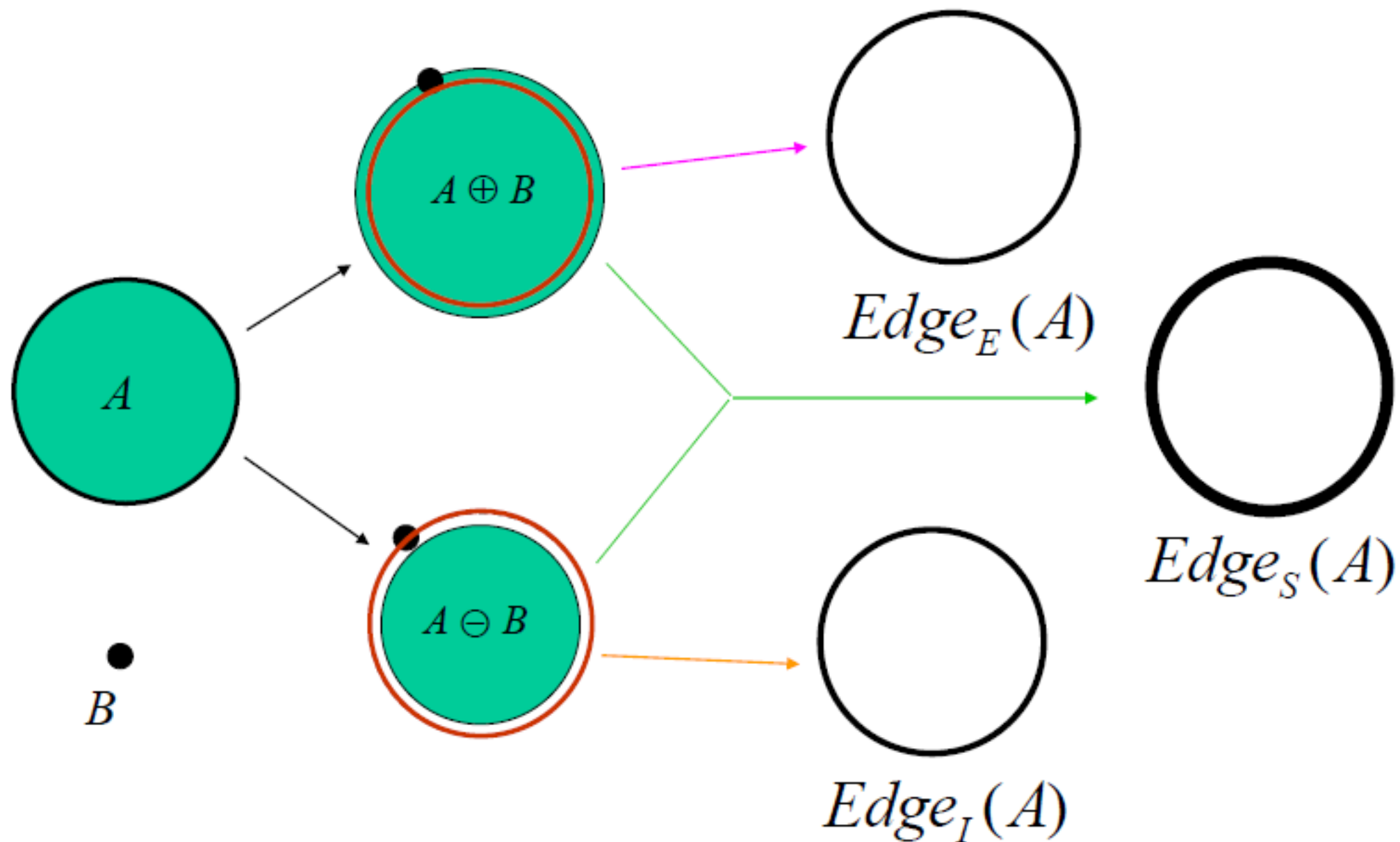
- Štandardné  $Edge_s(A) = (A \oplus B) - (A \ominus B)$

- Externé  $Edge_E(A) = (A \oplus B) - A$

- Interné  $Edge_I(A) = A - (A \ominus B)$

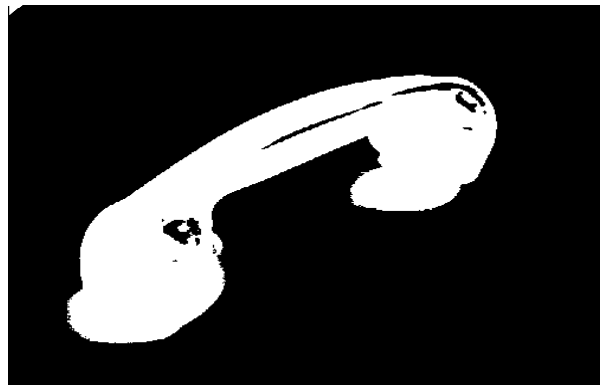


# Hľadanie hranice

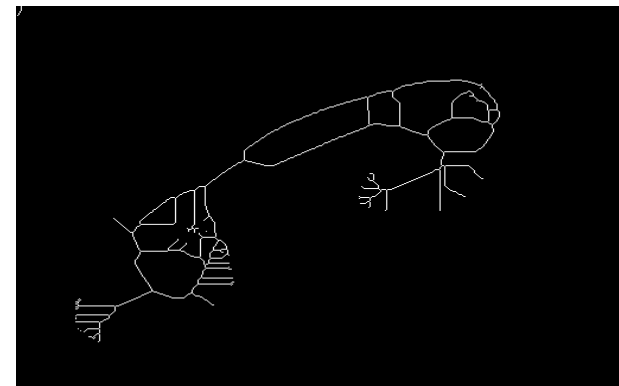


# Príklad využitia

prahovanie



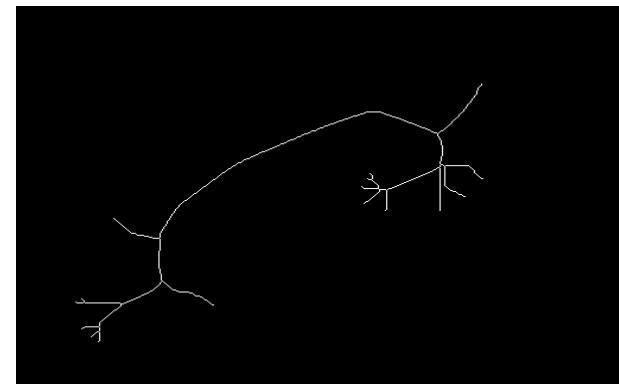
kostra



prahovanie



kostra



+ uzavretie



# MATLAB

**Erózia:** `imerode(Obrázok, štrukt. Element)`

**Dilatácia:** `imdilate(Obrázok, štrukt. Element)`

**Opening:** `imopen(Obrázok, štrukt. Element)`

**Closing:** `imclose(Obrázok, štrukt. Element)`

# MATLAB

Štruktúrálňy element:

SE = strel(shape, parameters);

**shape:**'arbitrary';'pair';'diamond';'periodicline'

'disk';'rectangle';'line';'square';'octagon'

se1 = strel('square',11)    % 11-by-11 square

se2 = strel('line',10,45)    % length 10, angle 45

se3 = strel('disk',15)    % disk, radius 15

se4 = strel('ball',15,5)    % ball, radius 15, height 5

# MATLAB

**Erózia:** `imerode(Obrázok, štrukt. Element)`

Obrázok **binárny** alebo **šedoúrovňový**

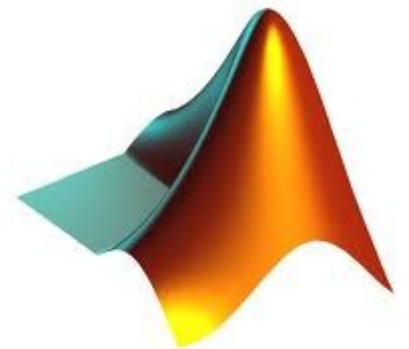
```
I = imread('circles.png');
```

```
%rgb2gray(); ak treba
```

```
se = strel('disk',11);
```

```
erl = imerode(I,se);
```

```
imshow(I); figure,imshow(erl);
```



# MATLAB

Ďalšie metódy

**Top Hat:** `imtophat(I,se);`

original I - opening;

**Bottom Hat:** `imbothat(I,se);`

closing- Original I;

**Hmax:** `imhmax(I,h)`

potlačí maximá menšie ako h

`imregionalmax(I); imhmin(I,h)...`

# MATLAB

## **Vlastnosti oblastí v Binárnom obr.**

`s = regionprops(Bin. obrazok, 'vlastnost');`

**vlastnosti:** 'Area', 'BoundingBox',

'Centroid', 'Orientation', 'Perimeter', 'ConvexArea'...

# MATLAB

```
bw = imread('text.png');  
L = bwlabel(bw);  
s = regionprops(L, 'centroid');  
centroids = cat(1, s.Centroid);  
imshow(bw)  
hold on  
plot(centroids(:,1), centroids(:,2), 'b*')  
hold off
```