



KATEDRA  
INFORMATIKY  
UNIVERZITA PALACKÉHO V OLOMOUCI

# Ořezávání dvourozměrných objektů

## Počítačová grafika

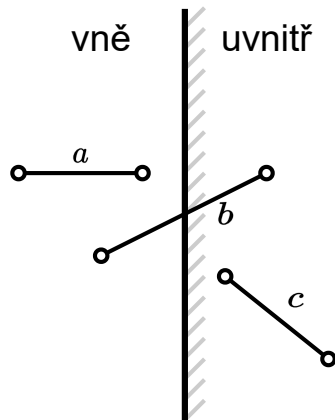
Mgr. Markéta Trnečková, Ph.D.

# Oblast

- Ořezávání objektů – **clipping**
- **Ořezová oblast:**
  - obdélníková – levý horní roh  $[xw_{min}, yw_{min}]$  a pravý dolní roh  $[xw_{max}, yw_{max}]$
  - obecná
- **Algoritmy:**
  - Test polohy bodu
  - Ořezání úsečky

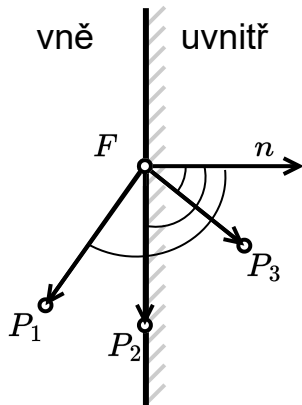
# Test polohy bodu

- Poloha bodu  $P = [p_x, p_y]$  vůči hraniční přímce  $F$
- $F(x, y) : ax + by + c = 0$ 
  - $F(p_x, p_y) > 0$  – bod leží uvnitř
  - $F(p_x, p_y) < 0$  – bod leží vně
  - $F(p_x, p_y) = 0$  – bod leží na hraniční přímce



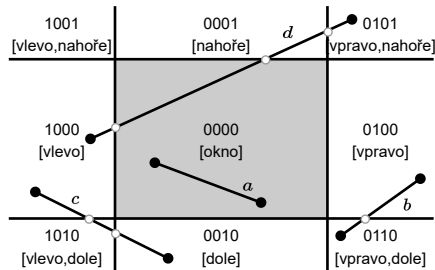
# Test polohy bodu

- bod na přímce  $F = [f_x, f_y]$
- normálový vektor  $\vec{n}$ 
  - $\vec{n} \cdot (P - F) < 0$  – tupý úhel, vně
  - $\vec{n} \cdot (P - F) > 0$  – ostrý úhel, uvnitř
  - $\vec{n} \cdot (P - F) = 0$  – bod leží na hraniční přímce



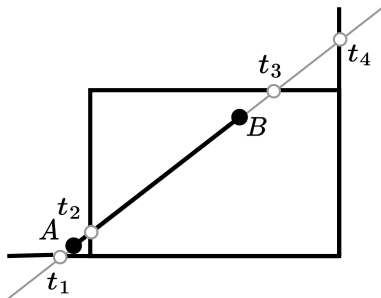
# Ořezání úsečky

- ořezávání obdélníkovým oknem
- **hraniční kód** – 4 bitová informace popisující polohu bodu vůči oknu
- **Určení polohy úsečky vůči bodu** ( $kod(P)$ ):
  - $kod(P_1) \vee kod(P_2) = 0000$  – celá úsečka leží v okně
  - $kod(P_1) \wedge kod(P_2) \neq 0000$  – celá úsečka mimo okno
  - $kod(P_1) \wedge kod(P_2) = 0000$  – úsečka prochází více oblastmi



# Cyrus-Beck algoritmus

- ořezávání libovolnou konvexní ořezovou oblastí
- Parametrické vyjádření úsečky ( $t \in \langle 0, 1 \rangle$ )  
 $P(t) = P_1 + (P_2 - P_1)t$
- **Průsečík úsečky s hraniční přímkou**  
 $t = \frac{\vec{n} \cdot (P_1 - F)}{\vec{n} \cdot (P_2 - P_1)}$
- Váhový vektor:  $\vec{w} = P_1 - F$
- Směrový vektor přímky:  $\vec{d} = P_2 - P_1$
- $\vec{w} \cdot \vec{n} \neq 0 \rightarrow$  existuje průsečík s hraniční přímkou
- Po ořezání úsečky se upraví  
 $t \in \langle t_{min}, t_{max} = 1 \rangle$



# Cyrus-Beck algoritmus

Úprava proměnných  $t_{min}$  a  $t_{max}$ :

1  $t_{min} = 0, t_{max} = 1, \vec{d} = P(1) - P(0)$

2 Pro všechny hranice  $i$  (vnitřní normála  $\vec{n}_i$ ):

1 Pokud  $\vec{d} \cdot \vec{n}_i \neq 0$ , pak  $t = -(\vec{w}_i \cdot \vec{n}_i) / \vec{d} \cdot \vec{n}_i$ .

Jinak nemá přímka s touto hraniční přímkou průsečík (konec).

2 Pokud platí:

■  $(\vec{d} \cdot \vec{n}_i > 0)$  a zároveň  $(t \leq 1) \rightarrow t_{min} = \max(t, t_{min})$

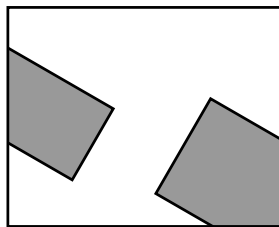
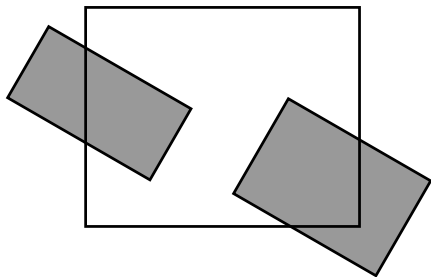
■  $(\vec{d} \cdot \vec{n}_i < 0)$  a zároveň  $(t \geq 0) \rightarrow t_{max} = \min(t, t_{max})$

3 Pokud platí  $\vec{w}_i \cdot \vec{n}_i < 0$  – úsečka je mimo (konec).

3 Pokud  $t_{min} < t_{max}$ , pak je výsledkem ořezaná úsečka  $P(t_{min})P(t_{max})$

## Ořezání polygonu

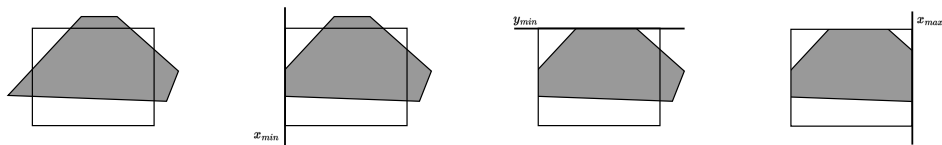
- ořezávání každé hraniční úsečky zvlášť – rozpad oblasti



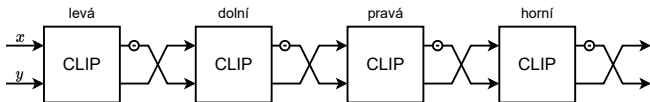


# Ořezání polygonu

- ořezávání postupně všemi hraničními přímkami

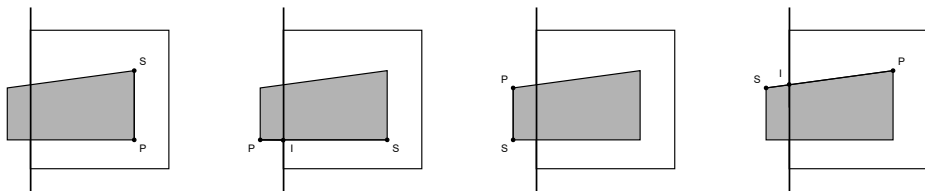


- ořezávání jednou úsečkou a otáčení oblasti



# Sutherland-Hodgmanův algoritmus

- postupně ořezáváme jednotlivé úsečky polygonu všemi ořezovými přímkami
- zpracováváme vrchol polygonu  $P$ , který spolu s předchozím vrcholem  $S$  tvoří úsečku



- 1 úsečka je celá uvnitř ořezového okna;  $P$  je předán další ořezové přímce ke zpracování
- 2 místo bodu  $P$  je předán další ořezové přímce bod  $I$
- 3 nepředáváme další ořezové přímce žádný bod
- 4 předáváme další přímce body  $I$  a  $P$

# Sutherland-Hodgmanův algoritmus

**1** bod  $S = [x_S, y_S]$  = poslední vrchol polygonu

**2** pro vrcholy  $P_i = [x_{P_i}, y_{P_i}]$  ( $i = 1, \dots, n$ ):

**2a** pokud  $x_{P_i} > x_{W_{min}}$ :

■ pokud  $x_S > x_{W_{min}}$ :

   přidej  $P_i$  další hranici k ořezání

■ jinak:

   vypočti průsečík  $I = [x_{W_{min}}, y_S + (x_{W_{min}} - x_S) \frac{y_{P_i} - y_S}{x_{P_i} - x_S}]$ .

   předej  $I$  další hranici k ořezání

   předej  $P_i$  další hranici k ořezání

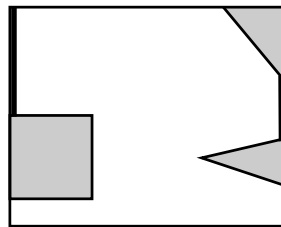
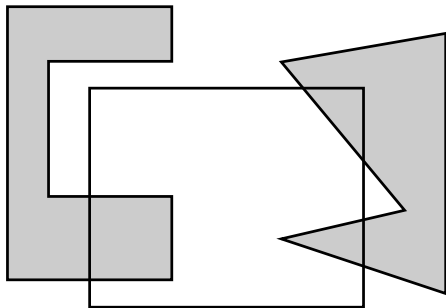
**2b** jinak pokud  $x_S > x_{W_{min}}$

■ vypočti průsečík  $I = [x_{W_{min}}, y_S + (x_{W_{min}} - x_S) \frac{y_{P_i} - y_S}{x_{P_i} - x_S}]$ .

■ předej  $I$  další hranici k ořezání

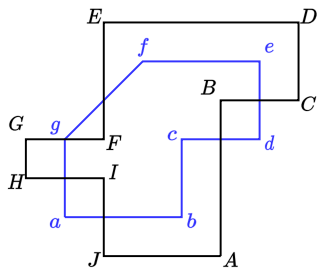
**2c** aktualizuj  $S$  jako  $P_i$

## Ořezávání nekonvexních polygonů

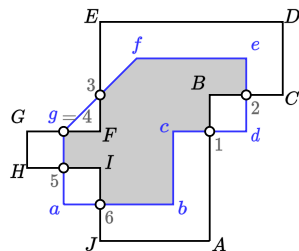
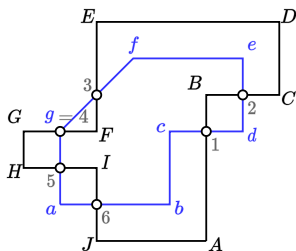
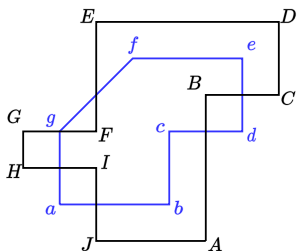


# Algoritmus Weiler-Atherton

- ořezávání nekonvexním oknem
- okno i polygon – seznam orientovaných uzavřených posloupností hran tvořících hranici
- **okno**:  $a - b - c - d - e - f - g$  (seznam  $W$ )
- **polygon**:  $A - B - C - D - E - F - G - H - I - J$  (seznam  $P$ )

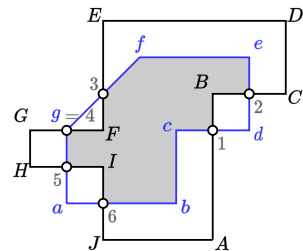
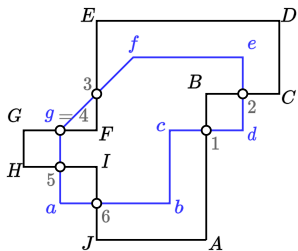
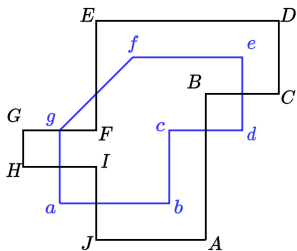


# Algoritmus Weiler-Atherton



- 1 H ← průsečíky mezi hranicemi polygonu a okna
- 2 zařaď průsečíky mezi vrcholy v seznamech P a W a propoj obousměrnými ukazateli
- 3 dokud H není prázdný opakuj:
  - 1 vyjmi průsečík ze seznamu H.
  - 2 přejdi do seznamu P nebo W
  - 3 dokud není uzavřen tah návratem do výchozího průsečíku opakuj:
    - přidej na výstup všechny vrcholy ze seznamu (P nebo W), které jsou před dalším průsečíkem
    - přejdi do druhého seznamu (P nebo W)

# Algoritmus Weiler-Atherton



**Výsledek:** 1 – B – 2 – e – f – 3 – F – 4 – 5 – I – 6 – b – c – 1