

Bevezetés a számítógépi grafikába

Vágóalgoritmusok

Troll Ede Mátyás

Matematikai és Informatikai Intézet
Eszterházy Károly Katolikus Egyetem

Eger, 2024



Áttekintés

- 1 Cohen-Sutherland vágó algoritmus
- 2 Szakasz vágása poligonra
 - Szakasz vágása konvex poligonra
 - Szakasz vágása konkáv poligonra
- 3 Poligon vágása téglalapra

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek
 - Téglalap

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek
 - Téglalap
 - Konvex poligon

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek
 - Téglalap
 - Konvex poligon
 - Konkáv poligon

A vágóalgoritmusok feladata

- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek
 - Téglalap
 - Konvex poligon
 - Konkáv poligon
 - Akár lyukak és szigetek

A vágóalgoritmusok feladata

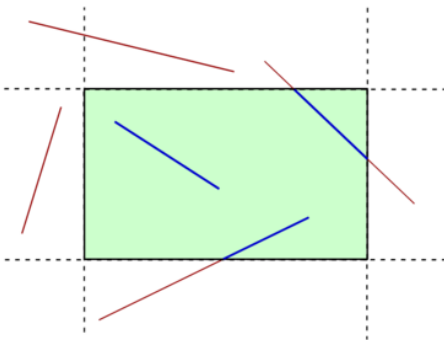
- A feladat a megrajzolandó szakaszok egy meghatározott ablakban történő megjelenítése
- Vágások végrehajtásával csak az ablakokban látható szakaszdarabokat fogjuk megjeleníteni
- Az ablakok lehetnek
 - Téglalap
 - Konvex poligon
 - Konkáv poligon
 - Akár lyukak és szigetek
 - Tetszőleges mélységben

Áttekintés

- 1 Cohen-Sutherland vágó algoritmus
- 2 Szakasz vágása poligonra
 - Szakasz vágása konvex poligonra
 - Szakasz vágása konkáv poligonra
- 3 Poligon vágása téglalapra

Cohen-Sutherland vágó algoritmus

Az algoritmus szakaszok téglalapra való vágását valósítja meg.



Cohen-Sutherland vágó algoritmus

Az algoritmust Danny Cohen és Ivan Sutherland fejlesztette ki
1967-ben egy repülő szimulátor részeként.



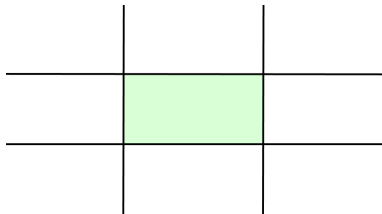
Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

A 9 síkrészhez 4 bites kódokat rendelünk az oldalegyenesekhez való viszonyuk alapján.



Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

A 9 síkrészhez 4 bites kódokat rendelünk az oldalegyenesekhez való viszonyuk alapján.

- Az 1. bit 1, ha a végpont a felső vízszintes vonal fölött van, egyébként 0

1	1	1
0	0	0
0	0	0

Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

A 9 síkrészhez 4 bites kódokat rendelünk az oldalegyenesekhez való viszonyuk alapján.

- Az 1. bit 1, ha a végpont a felső vízszintes vonal fölött van, egyébként 0
- A 2. bit 1, ha a végpont az alsó vízszintes vonal alatt van, egyébként 0

1 0	1 0	1 0
0 0	0 0	0 0
0 1	0 1	0 1

Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

A 9 síkrészhez 4 bites kódokat rendelünk az oldalegyenesekhez való viszonyuk alapján.

- Az 1. bit 1, ha a végpont a felső vízszintes vonal fölött van, egyébként 0
- A 2. bit 1, ha a végpont az alsó vízszintes vonal alatt van, egyébként 0
- A 3. bit 1, ha a végpont a jobb oldali függőleges vonaltól jobbra van, egyébként 0

1 0 0	1 0 0	1 0 1
0 0 0	0 0 0	0 0 1
0 1 0	0 1 0	0 1 1

Síktartományok kódjai

A téglalap 4 oldala segítségével a síkot 9 részre osztjuk.

A 9 síkrészhez 4 bites kódokat rendelünk az oldalegyenesekhez való viszonyuk alapján.

- Az 1. bit 1, ha a végpont a felső vízszintes vonal fölött van, egyébként 0
- A 2. bit 1, ha a végpont az alsó vízszintes vonal alatt van, egyébként 0
- A 3. bit 1, ha a végpont a jobb oldali függőleges vonaltól jobbra van, egyébként 0
- A 4. bit 1, ha a végpont a bal oldali függőleges vonaltól bal van, egyébként 0

1 0 0 1	1 0 0 0	1 0 1 0
0 0 0 1	0 0 0 0	0 0 1 0
0 1 0 1	0 1 0 0	0 1 1 0

Síktartományok kódjai

```
FÜGGVÉNY VEGPONT_KOD(PONT P): EGÉSZ;  
VÁLTOZÓK  
  EGÉSZ: KOD;  
ALGORITMUS  
  KOD <- 0;  
  HA (P.X < ABLAK.BAL) AKKOR  
    KOD <- KOD VAGY 1;  
  KÜLÖNBEN  
    HA (P.X > ABLAK.JOBB) AKKOR  
      KOD <- KOD VAGY 2;  
    HA_VÉGE;  
  HA_VÉGE;  
  
  HA (P.Y < ABLAK.LENT) AKKOR  
    KOD <- KOD VAGY 4;  
  KÜLÖNBEN  
    HA (P.Y > ABLAK.FENT) AKKOR  
      KOD <- KOD VAGY 8;  
    HA_VÉGE;  
  HA_VÉGE;  
  
  VEGPONT_KOD <- KOD;  
FÜGGVÉNY_VÉGE;
```

1 0 0 1	1 0 0 0	1 0 1 0
0 0 0 1	0 0 0 0	0 0 1 0
0 1 0 1	0 1 0 0	0 1 1 0

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert
 - az algoritmus felépítése miatt nem vezethet 0-val való osztáshoz

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert
 - az algoritmus felépítése miatt nem vezethet 0-val való osztáshoz
 - a szakaszrajzoló fel van készítve erre az esetre.

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert
 - az algoritmus felépítése miatt nem vezethet 0-val való osztáshoz
 - a szakaszrajzoló fel van készítve erre az esetre.
- Ha a két kódnak **azonos helyen 1-es értéke** van, akkor **a két pont ugyanazon oldalegyenesen kívülre esik**, tehát befejezhetjük az algoritmust rajzolás nélkül.

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert
 - az algoritmus felépítése miatt nem vezethet 0-val való osztáshoz
 - a szakaszrajzoló fel van készítve erre az esetre.
- Ha a két kódnak **azonos helyen 1-es értéke** van, akkor **a két pont ugyanazon oldalegyenesen kívülre esik**, tehát befejezhetjük az algoritmust rajzolás nélkül.
- Ellenkező esetben megvizsgáljuk, hogy **a végpontok csupa 0 kóddal rendelkeznek**-e. Ebben az esetben **mindkét pont az ablak belső pontja**, így megrajzolhatjuk a szakaszt.

Cohen-Sutherland vágó algoritmus

- Legyen adott a szakasz a $\mathbf{p}_0(x_0; y_0)$ és $\mathbf{p}_1(x_1; y_1)$ végpontjaival
- Végpontok egyezőségét nem kell ellenőrizni, mert
 - az algoritmus felépítése miatt nem vezethet 0-val való osztáshoz
 - a szakaszrajzoló fel van készítve erre az esetre.
- Ha a két kódnak **azonos helyen 1-es értéke** van, akkor **a két pont ugyanazon oldalegyenesen kívülre esik**, tehát befejezhetjük az algoritmust rajzolás nélkül.
- Ellenkező esetben megvizsgáljuk, hogy **a végpontok csupa 0 kóddal rendelkeznek-e**. Ebben az esetben **mindkét pont az ablak belső pontja**, így megrajzolhatjuk a szakaszt.
- Amennyiben különböző helyeken 1-es érték szerepel, úgy vágást kell végrehajtanunk.

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

$$-(y_1 - y_0)x + (x_1 - x_0)c = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

$$-(y_1 - y_0)x + (x_1 - x_0)c = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

$$-(y_1 - y_0)x = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (x_1 - x_0)c$$

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

$$-(y_1 - y_0)x + (x_1 - x_0)c = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

$$-(y_1 - y_0)x = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (x_1 - x_0)c$$

$$(y_1 - y_0)x = (y_1 - y_0)x_0 + (x_1 - x_0)(c - y_0)$$

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

$$\begin{aligned} -(y_1 - y_0)x + (x_1 - x_0)c &= -(y_1 - y_0)x_0 + (x_1 - x_0)y_0 \\ -(y_1 - y_0)x &= -(y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (x_1 - x_0)c \\ (y_1 - y_0)x &= (y_1 - y_0)x_0 + (x_1 - x_0)(c - y_0) \\ x &= \frac{(y_1 - y_0)x_0 + (x_1 - x_0)(c - y_0)}{(y_1 - y_0)} \end{aligned}$$

Vágás vízszintes vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás vízszintes él esetén:

$$h : y = c, \text{ ahol } c \in \{y_{min}, y_{max}\}$$

$$e : -(y_1 - y_0)x + (x_1 - x_0)y = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

Behelyettesítés után

$$-(y_1 - y_0)x + (x_1 - x_0)c = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0$$

$$-(y_1 - y_0)x = -(y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (x_1 - x_0)c$$

$$(y_1 - y_0)x = (y_1 - y_0)x_0 + (x_1 - x_0)(c - y_0)$$

$$x = \frac{(y_1 - y_0)x_0 + (x_1 - x_0)(c - y_0)}{(y_1 - y_0)}$$

$$x = x_0 + \frac{(x_1 - x_0)(c - y_0)}{(y_1 - y_0)}$$

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

$$(y_1 - y_0)c - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

$$(y_1 - y_0)c - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

$$- (x_1 - x_0)y = (y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (y_1 - y_0)c$$

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

$$\begin{aligned}(y_1 - y_0)c - (x_1 - x_0)y &= (y_1 - y_0)x_0 - (x_1 - x_0)y_0 \\ - (x_1 - x_0)y &= (y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (y_1 - y_0)c \\ (x_1 - x_0)y &= (x_1 - x_0)y_0 + (y_1 - y_0)(c - x_0)\end{aligned}$$

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

$$\begin{aligned}(y_1 - y_0)c - (x_1 - x_0)y &= (y_1 - y_0)x_0 - (x_1 - x_0)y_0 \\ - (x_1 - x_0)y &= (y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (y_1 - y_0)c \\ (x_1 - x_0)y &= (x_1 - x_0)y_0 + (y_1 - y_0)(c - x_0) \\ y &= \frac{(x_1 - x_0)y_0 + (y_1 - y_0)(c - x_0)}{(x_1 - x_0)}\end{aligned}$$

Vágás függőleges vonallal

Azt, hogy melyik oldallal kell vágni a szakaszt, a 4 bites kódban szereplő 1-es érték pozíciója árulja el.

Vágás függőleges él esetén:

$$h : x = c, \text{ ahol } c \in \{x_{min}, x_{max}\}$$

$$e : (y_1 - y_0)x - (x_1 - x_0)y = (y_1 - y_0)x_0 - (x_1 - x_0)y_0$$

Behelyettesítés után

$$\begin{aligned}(y_1 - y_0)c - (x_1 - x_0)y &= (y_1 - y_0)x_0 - (x_1 - x_0)y_0 \\ - (x_1 - x_0)y &= (y_1 - y_0)x_0 + (x_1 - x_0)y_0 - (y_1 - y_0)c \\ (x_1 - x_0)y &= (x_1 - x_0)y_0 + (y_1 - y_0)(c - x_0) \\ y &= \frac{(x_1 - x_0)y_0 + (y_1 - y_0)(c - x_0)}{(x_1 - x_0)} \\ y &= y_0 + \frac{(y_1 - y_0)(c - x_0)}{(x_1 - x_0)}\end{aligned}$$

Cohen-Sutherland vágó algoritmus

Vágás után a mozgatott végponthoz újra kiszámítjuk a kódot

Cohen-Sutherland vágó algoritmus

Vágás után a mozgatott végponthoz újra kiszámítjuk a kódot

Nem csak 1-es bit tűnhet el, hanem akár új is keletkezhet

Cohen-Sutherland vágó algoritmus

Vágás után a mozgatott végponthoz újra kiszámítjuk a kódot

Nem csak 1-es bit tűnhet el, hanem akár új is keletkezhet

Ezek után a kódok vizsgálatát előlről kezdjük

Cohen-Sutherland vágó algoritmus

Vágás után a mozgatott végponthoz újra kiszámítjuk a kódot

Nem csak 1-es bit tűnhet el, hanem akár új is keletkezhet

Ezek után a kódok vizsgálatát előlről kezdjük

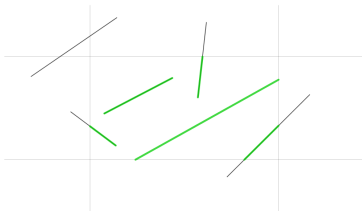
A megadott lépéseket követve véges sok lépés után meghatározhatjuk a szakasz látható darabját, vagy megmutathatjuk, hogy nincs látható része

Cohen-Sutherland vágó algoritmus

```

ELJÁRÁS COHEN_SUTH_VAZL(PONT P0, PONT P1)
VÁLTOZÓK
  EGÉSZ: KODO, KOD1;
  LOGIKAI: ELFOGAD;
ALGORITMUS
  KODO <- VEGPONT_KOD(P0);
  KOD1 <- VEGPONT_KOD(P1);
  ELFOGAD <- HAMIS;
  CIKLUS_AMÍG (IGAZ)
    HA ((KODO VAGY KOD1) = HAMIS) AKKOR
      ELFOGAD = IGAZ;
      CIKLUS_KILÉP;
    KÜLÖNBEN
      HA ((KODO ÉS KOD1) = IGAZ) AKKOR
        CIKLUS_KILÉP;
      KÜLÖNBEN
        AZ_EGYIK_KINTI_PONT_VÁGÁSA;
        A_PONTHOZ_TARTOZÓ_KÓD_ÚJRASZÁMÍTÁSA;
      HA_VÉGE
    HA_VÉGE;
  CIKLUS_VÉGE;
  HA (ELFOGAD = IGAZ) AKKOR
    SZAKASZ(P0, P1);
  HA_VÉGE;
ELJÁRÁS_VÉGE;

```



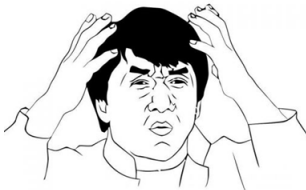
Cohen-Sutherland vágó algoritmus

Megjegyezzük, hogy az eljárás könnyen általánosítható **3 dimenzióra** is.

Cohen-Sutherland vágó algoritmus

Megjegyezzük, hogy az eljárás könnyen általánosítható **3 dimenzióra** is.

Ekkor a teret a vágó téglatest alapján **27 térrészre osztjuk**, és az egyes irányokhoz **6 bites kódokat** rendelünk.

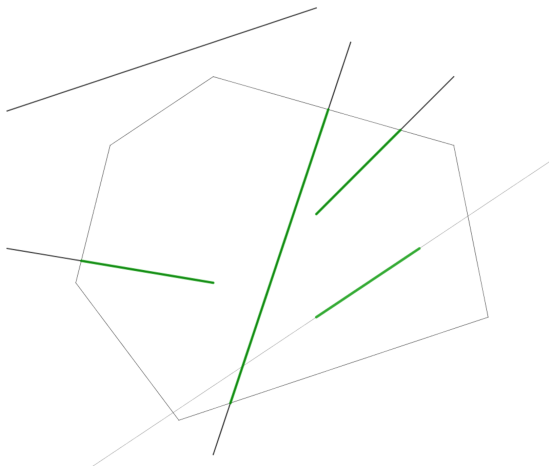


Áttekintés

- 1 Cohen-Sutherland vágó algoritmus
- 2 Szakasz vágása poligonra
 - Szakasz vágása konvex poligonra
 - Szakasz vágása konkáv poligonra
- 3 Poligon vágása téglalapra

Szakasz vágása konvex poligonra

Szakasz vágása konvex poligonra



Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét

Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét

Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik

Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Ha egymást követő elemek különböznek, akkor az adott szakaszon vágást kell végrehajtani, azaz ki kell számolni a két egyenes metszéspontját.

Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Ha egymást követő elemek különböznek, akkor az adott szakaszon vágást kell végrehajtani, azaz ki kell számolni a két egyenes metszéspontját.
- Mivel konvex alakzatról beszélünk, így 2 helyen történik vágás

Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Ha egymást követő elemek különböznek, akkor az adott szakaszon vágást kell végrehajtani, azaz ki kell számolni a két egyenes metszéspontját.
- Mivel konvex alakzatról beszélünk, így 2 helyen történik vágás
- Természetesen le kell kezelni azt az esetet, amikor a szakasz a poligonon belül van, mert ekkor az egyenesnek szintén van 2 metszéspontja, viszont hamis eredményt kapunk

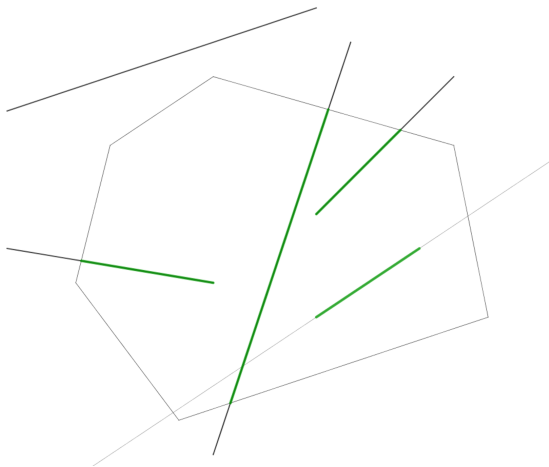
Szakasz vágása konvex poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Ha egymást követő elemek különböznek, akkor az adott szakaszon vágást kell végrehajtani, azaz ki kell számolni a két egyenes metszéspontját.
- Mivel konvex alakzatról beszélünk, így 2 helyen történik vágás
- Természetesen le kell kezelni azt az esetet, amikor a szakasz a poligonon belül van, mert ekkor az egyenesnek szintén van 2 metszéspontja, viszont hamis eredményt kapunk
- Ha kiszámoltuk a metszéspontokat, rendezzük azokat valamely (nem 0) koordináta szerint

Szakasz vágása konvex poligonra

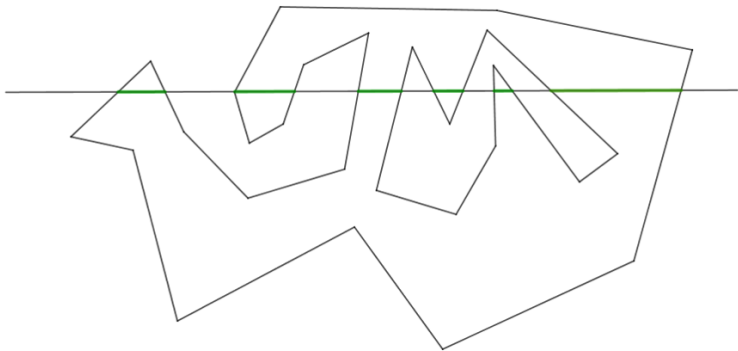
- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Ha egymást követő elemek különböznek, akkor az adott szakaszon vágást kell végrehajtani, azaz ki kell számolni a két egyenes metszéspontját.
- Mivel konvex alakzatról beszélünk, így 2 helyen történik vágás
- Természetesen le kell kezelni azt az esetet, amikor a szakasz a poligonon belül van, mert ekkor az egyenesnek szintén van 2 metszéspontja, viszont hamis eredményt kapunk
- Ha kiszámoltuk a metszéspontokat, rendezzük azokat valamely (nem 0) koordináta szerint
- Végül megrajzoljuk a kapott szakaszt a két középső pont között

Szakasz vágása konvex poligonra



Szakasz vágása konkáv poligonra

Szakasz vágása konkáv poligonra



Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét
- Kiszámoljuk a metszéspontokat

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét
- Kiszámoljuk a metszéspontokat
- A metszéspontokat x , vagy y koordináta szerint rendezzük

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét
- Kiszámoljuk a metszéspontokat
- A metszéspontokat x , vagy y koordináta szerint rendezzük
 - Ha az egyenes párhuzamos valamely tengellyel, akkor az alapján a koordináta szerint rendezünk

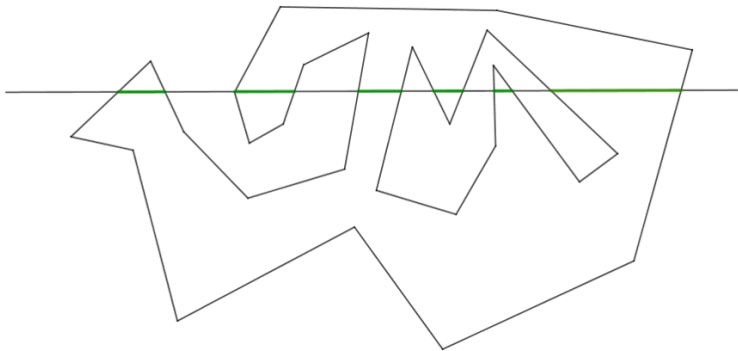
Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét
- Kiszámoljuk a metszéspontokat
- A metszéspontokat x , vagy y koordináta szerint rendezzük
 - Ha az egyenes párhuzamos valamely tengellyel, akkor az alapján a koordináta szerint rendezünk
- Beszúrjuk a szakasz két végpontját a rendezett sorba és megnézzük, hogy hol van a két végpont a metszéspontokhoz képest

Szakasz vágása konkáv poligonra

- A vizsgált szakasz végpontjai alapján felírjuk a vizsgált egyenes egyenletét
- Ebbe sorra behelyettesítjük az alakzat koordinátáit, és tároljuk az eredmények előjelét
 - Ha minden esetben azonos az előjel, akkor a szakasz a poligonon kívülre esik
 - Előjelváltás esetén az adott poligon él egyenese metszi a szakasz egyenesét
- Kiszámoljuk a metszéspontokat
- A metszéspontokat x , vagy y koordináta szerint rendezzük
 - Ha az egyenes párhuzamos valamely tengellyel, akkor az alapján a koordináta szerint rendezünk
- Beszúrjuk a szakasz két végpontját a rendezett sorba és megnézzük, hogy hol van a két végpont a metszéspontokhoz képest
- A szakasz egyik végpontjától indulva a páratlan és páros sorszámú metszéspontok közötti szakaszokat kell rajzolni a két végpont között

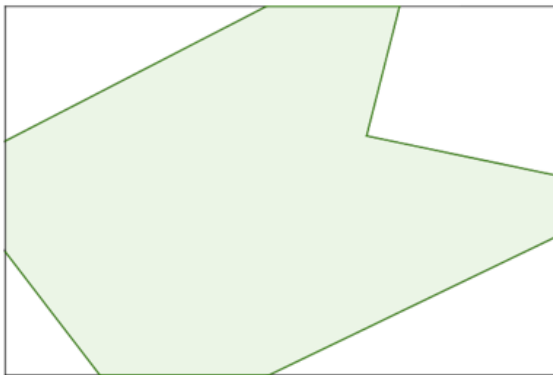
Szakasz vágása konkáv poligonra



Áttekintés

- 1 Cohen-Sutherland vágó algoritmus
- 2 Szakasz vágása poligonra
 - Szakasz vágása konvex poligonra
 - Szakasz vágása konkáv poligonra
- 3 Poligon vágása téglalapra

Poligon vágása téglalapra



Sutherland – Hodgman poligon vágó algoritmus téglalapra

- Vegyük figyelembe, hogy nem elég pusztán a poligon éleit vágnunk, mert úgy csak az élek egy halmazát kapjuk

Sutherland – Hodgman poligon vágó algoritmus téglalapra

- Vegyük figyelembe, hogy nem elég pusztán a poligon éleit vágnunk, mert úgy csak az élek egy halmazát kapjuk
 - Ez nem poligon!

Sutherland – Hodgman poligon vágó algoritmus téglalapra

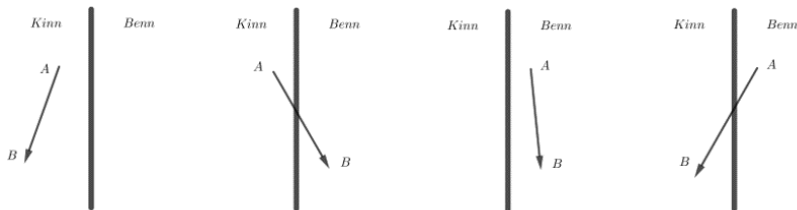
- Vegyük figyelembe, hogy nem elég pusztán a poligon éleit vágnunk, mert úgy csak az élek egy halmazát kapjuk
 - Ez nem poligon!
- Az ablak négy élével sorra elvágjuk az alakzatot

Sutherland – Hodgman poligon vágó algoritmus téglalapra

- Vegyük figyelembe, hogy nem elég pusztán a poligon éleit vágnunk, mert úgy csak az élek egy halmazát kapjuk
 - Ez nem poligon!
- Az ablak négy élével sorra elvágjuk az alakzatot
- Az ablak éleire vonatkozóan létrehozunk egy-egy listát, melyben az eredeti csúcsok sorozata van a következők szerint

Sutherland – Hodgman poligon vágó algoritmus téglalapra

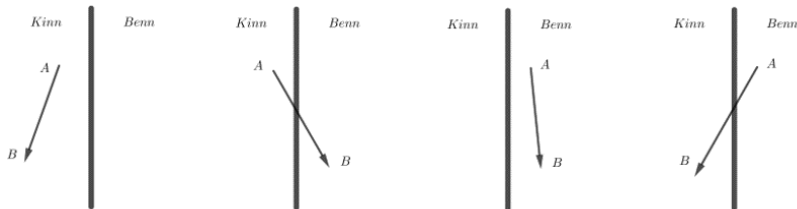
A poligon minden AB (irányított) élére az alábbi esetek lehetősége áll fenn



Sutherland – Hodgman poligon vágó algoritmus téglalapra

A poligon minden AB (irányított) élére az alábbi esetek lehetősége áll fenn

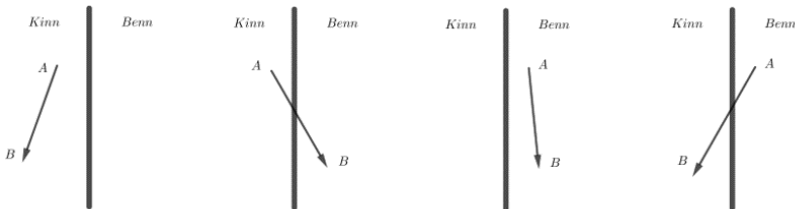
- Mindkét csúcs kívül van, nincs metszet



Sutherland – Hodgman poligon vágó algoritmus téglalapra

A poligon minden AB (irányított) élére az alábbi esetek lehetősége áll fenn

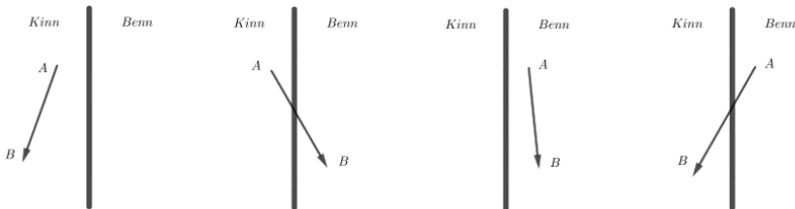
- Mindkét csúcs kívül van, nincs metszet
- B benn van, A kinn. Ekkor az él és az AB metszéspontja, majd B is felkerül a listára



Sutherland – Hodgman poligon vágó algoritmus téglalapra

A poligon minden AB (irányított) élére az alábbi esetek lehetősége áll fenn

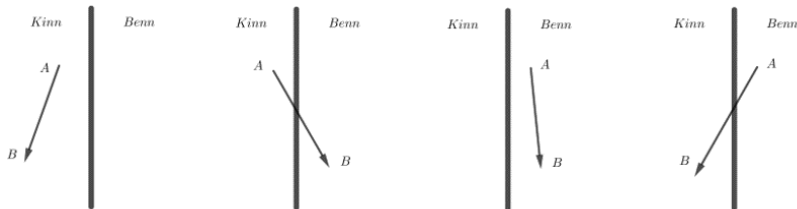
- Mindkét csúcs kívül van, nincs metszet
- B benn van, A kinn. Ekkor az él és az AB metszéspontja, majd B is felkerül a listára
- Mindkét csúcs belül van. Ekkor B felkerül a listára (Ha nem az első élről van szó, akkor A már rajta van)



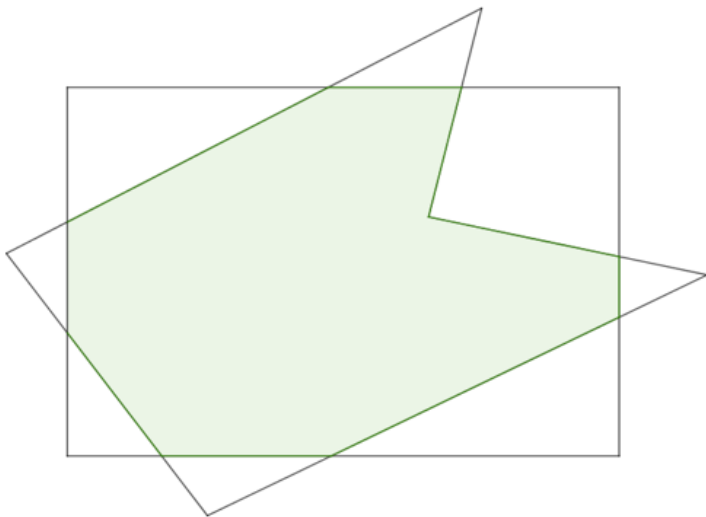
Sutherland – Hodgman poligon vágó algoritmus téglalapra

A poligon minden AB (irányított) élére az alábbi esetek lehetősége áll fenn

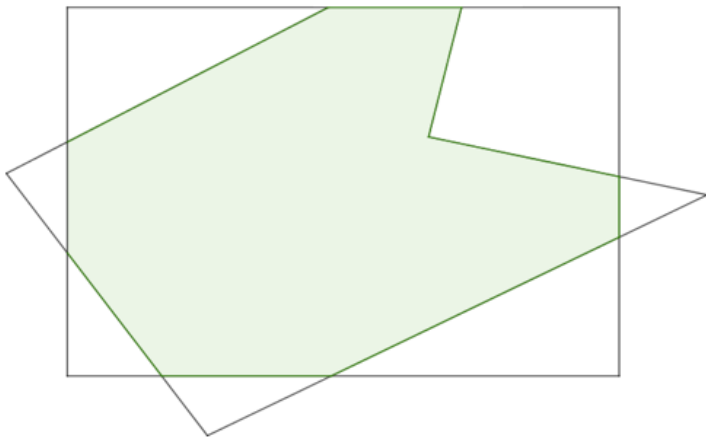
- Mindkét csúcs kívül van, nincs metszet
- B benn van, A kinn. Ekkor az él és az AB metszéspontja, majd B is felkerül a listára
- Mindkét csúcs belül van. Ekkor B felkerül a listára (Ha nem az első élről van szó, akkor A már rajta van)
- A benn van, B kinn. Az ablak éle és AB metszéspontja felkerül a listára



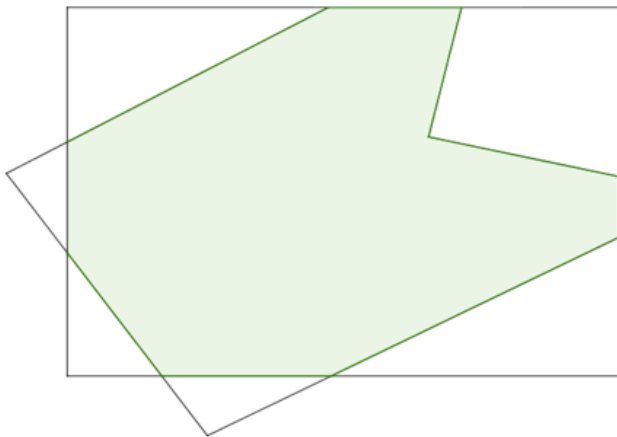
Sutherland – Hodgman poligon vágó algoritmus téglalapra



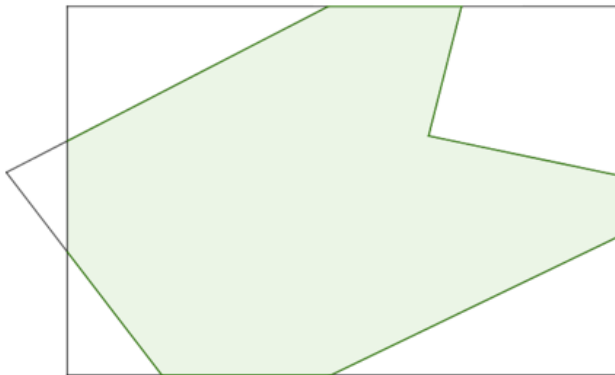
Sutherland – Hodgman poligon vágó algoritmus téglalapra



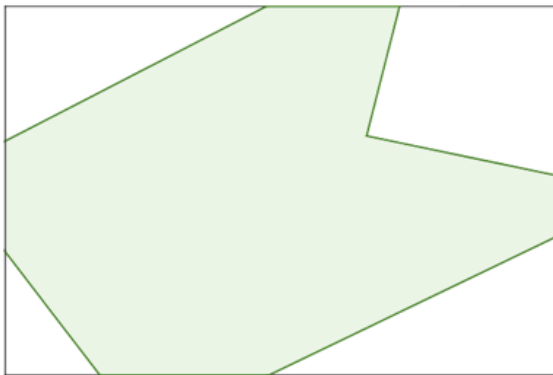
Sutherland – Hodgman poligon vágó algoritmus téglalapra



Sutherland – Hodgman poligon vágó algoritmus téglalapra



Sutherland – Hodgman poligon vágó algoritmus téglalapra



Köszönöm a figyelmet!