

*FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA SVEUČILIŠTA U ZAGREBU*

*Zavod za elektroniku, mikroelektroniku,  
računalne i inteligentne sustave*

## **INTERAKTIVNA RAČUNALNA GRAFIKA**

Laboratorijske vježbe

*Željka Mihajlović*

*Zagreb, 2016.*

## UVOD

Laboratorijske vježbe sastavni su dio izučavanja gradiva kolegija interaktivne računalne grafike. Praktičnim radom u laboratoriju usvaja se, utvrđuje i proširuje znanje iz računalne grafike. Ujedno tim se putem stiče osjećaj težine pojedinih postupaka.

Osnovne ideje temeljnih postupaka potrebno je razumjeti i praktičnim putem usvojiti, kako bi temeljni princip bio upotrebljiv i prilikom rješavanja problema u nekom drugom okruženju. Gotovo u svakom postupku postoje i posebni slučajevi o kojima je potrebno posebno voditi računa, a taj dio se uglavnom ostavlja kao dodatni izazov istraživanja.

Izabrani dijelovi gradiva uključeni su u laboratorijske vježbe. Prvenstveno, tu se radi o:

- grafičkim primitivima,
- geometrijskim izračunavanjima,
- transformacijama i projekcijama,
- izračunavanje osvjetljenja,
- postupci sjenčanja,
- skrivenim linijama i površinama,
- postupcima interpolacije prostorne krivulje,
- fraktalnim skupovima.

U svakoj vježbi potrebno je izraditi radni zadatak, odnosno radni program. Uz radni zadatak, u obliku podsjetnika, sažeto je ponovljeno nastavno gradivo. Također je naznačeno moguće rješenje radnog zadatka, u obliku postupka, tj. specifikacije radnog programa, te očekivani rezultati izvođenja radnog programa.

Izrada radnog programa zahtijeva skromnu sklopovsku i programsku opremu u laboratoriju. Traži se oprema koja daje radnom programu mogućnost iscrtavanja točke, povlačenja linije, promjenu boje na zaslonu i iscrtavanje trokuta s različitim intenzitetom boje u vrhovima.

Prva vježba daje osnovne matematičke operacije koje se koriste u računalnoj grafici. To su na primjer skalarni produkt vektora, vektorski produkt vektora, množenje matrica. Druga vježba je iscrtavanje linije Bresenhamovim postupkom. Ova vježba daje temeljni postupak pretvorbe iz kontinuiranog oblika u diskretni oblik potreban prilikom prikaza na zaslonu.

U trećoj vježbi upotrebljava se algoritam za popunjavanje konveksnog poligona i obavljaju se neka osnovna geometrijska izračunavanja prilikom ispitivanja odnosa točke i poligona u prostoru. Popunjavanje poligona temelji se na izračunavanju sjecišta pravaca. U četvrtoj vježbi potrebno je izgraditi topološku strukturu tijela i ispitati odnos točke i tijela.

Peta vježba opisuje postupak transformacije pogleda i perspektivnu projekciju, te se koristi za projiciranje izgrađenog tijela. Izrada prostorne krivulje postupkom Bezijera opisana je u šestoj vježbi, a načinjena krivulja se koristi za ostvarivanje animacije prilikom prikazivanja tijela u žičnoj formi.

Kako bi izgrađeno tijelo u prethodnim vježbama bilo vjernije prikazano, potrebno ga je osjenčati. Sedma vježba daje usporedbu konstantnog sjenčanja i Gouraud-ovog sjenčanja, na osnovi proračuna intenziteta, te postupak uklanjanja stražnjih poligona. Osmu vježbu daje jednostavan postupak za izradu i prikaz fraktalnih skupova.

## 1. Osnovne matematičke operacije u računalnoj grafici

Linearna algebra dio je matematike koji se bavi vektorima i matricama. U računalnoj grafici želimo prikazati objekte čiju površinu aproksimiramo točkama i poligonima. Pogodan način za zapis ovakvih podataka su vektori, a za zapis transformacija nad objektima kao što je pomicanje ili translacija, rotacija i slično pogodan način zapisa su matrice. Za to će nam biti potrebne osnovne matrice operacije što je potrebno ostvariti u ovoj vježbi. Poželjno je imati biblioteku osnovnih matematičkih operacija koja će biti korištena u daljnjim vježbama.

Odabir programskog jezika i programska implementacija koju je potrebno načiniti u vježbi, odnosno način na koji će biti definirane strukture podataka i sama implementacija u potpunosti se prepušta izboru studenta. Dozvoljeno je korištenje i već gotovih matematičkih biblioteka kao što je na primjer glm *OpenGL Mathematics* (<http://glm.g-truc.net/>) ili bilo koje slične biblioteke a u tom slučaju potrebno je demonstrirati njihovo korištenje.

Za razumijevanje složenijih postupaka osnovu čini razumijevanje elementarnih pojmova kao i geometrijska interpretacija osnovnih operacija.

### 1.1. Operacije s vektorima

Za zadana dva vektora, na primjer vektori  $v_1 = (x_1 \ y_1 \ z_1)$  i  $v_2 = (x_2 \ y_2 \ z_2)$  definiran je njihov zbroj i razlika.

$$v_1 + v_2 = (x_1 + x_2 \ y_1 + y_2 \ z_1 + z_2), \quad (1.1)$$

$$v_1 - v_2 = (x_1 - x_2 \ y_1 - y_2 \ z_1 - z_2).$$

Skalarni produkt vektora  $V_1 = (x_1 \ y_1 \ z_1)$  i  $V_2 = (x_2 \ y_2 \ z_2)$  računa se kao zbroj umnožaka pojedinih komponenti, a rezultat je skalar:

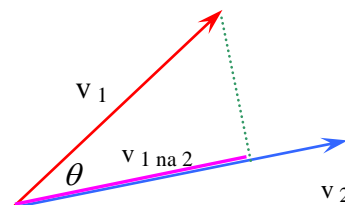
$$v_1 \cdot v_2 = x_1 x_2 + y_1 y_2 + z_1 z_2. \quad (1.2)$$

Skalarni produkt vektora koristi se kod izračuna kuta  $\theta$  između dva vektora prema izrazu:

$$\cos(\theta) = \frac{v_1 \cdot v_2}{|v_1| |v_2|} = \frac{x_1 x_2 + y_1 y_2 + z_1 z_2}{\sqrt{x_1^2 + y_1^2 + z_1^2} \sqrt{x_2^2 + y_2^2 + z_2^2}}. \quad (1.3)$$

gdje su u nazivniku moduli pojedinih vektora, odnosno njihove duljine.

Vrlo korisna je znati kako odrediti projekciju jednog vektora na drugi. Projekcija vektora  $v_1 = (x_1 \ y_1 \ z_1)$  na vektor  $v_2 = (x_2 \ y_2 \ z_2)$  je  $v_{1 \text{ na } 2}$ .



Slika 1.1. Projekcija vektora  $v_1$  na vektor  $v_2$ .

Kod projekcije jednog vektora na drugi možemo promatrati skalarnu vrijednost, odnosno duljinu projiciranog vektora  $v_1$  na vektor  $v_2$

$$v_{1na2} = \cos(\theta) |v_1| = \frac{v_1 \cdot v_2}{|v_2|} \quad (1.4)$$

do čega dolazimo iz izraza (1.3) i Slike 1.1. uz interpretaciju definicije kosinusa kuta kao omjera priležeće stranice trokuta ( $v_{1na2}$ ) i hipotenuze trokuta ( $|v_1|$ ). Znači, projekciju jednog vektora na drugi možemo odrediti iz poznavanja tih vektora ili kuta i vektora kojeg projiciramo.

Ako nam je potreban vektor koji predstavlja projekciju  $v_1$  na vektor  $v_2$  prethodno dobiveni rezultat ćemo pomnožiti jediničnim vektorom u smjeru vektora  $v_2$ .

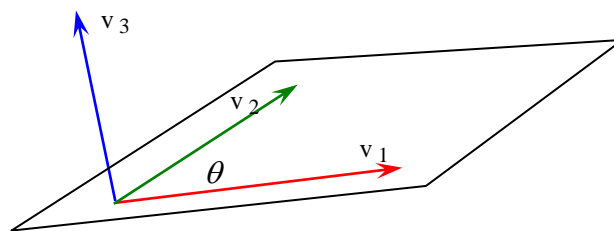
$$v_{1na2} = \cos(\theta) |v_1| \frac{v_2}{|v_2|} = \frac{v_1 \cdot v_2}{|v_2|} \frac{v_2}{|v_2|} \quad (1.5)$$

Sada je rezultat vektor.

Vektorski produkt vektora  $v_1 = (x_1 \ y_1 \ z_1)$  i  $v_2 = (x_2 \ y_2 \ z_2)$  rezultira vektorom i definiran je

$$v_1 \times v_2 = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_1 & y_1 & h_1 \\ x_2 & y_2 & h_2 \end{bmatrix} = \begin{bmatrix} y_1 h_2 - y_2 h_1 \\ -x_1 h_2 + x_2 h_1 \\ x_1 y_2 - x_2 y_1 \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (1.6)$$

Geometrijska interpretacija vektorskog produkta dva vektora je vektor koji je okomit na ravninu u kojoj leže vektori  $v_1$  i  $v_2$ . Važan je i redoslijed ova dva vektora u vektorskom produktu a definiran je po pravilu desne ruke. Ako zakretanje vektora  $v_1$  prema  $v_2$  određuju prsti desne ruke, palac je u smjeru resultantnog vektora  $v_3$ .



Slika 1.2. Vektorski produkt vektora  $v_1$  i  $v_2$  daje vektor  $v_3$ .

Ako zamijenimo redoslijed vektora  $v_1$  i  $v_2$  u vektorskom produktu, resultantni vektor  $v_3$  bit će usmjeren na drugu stranu ravnine.

## 1.2. Operacije s matricama

U računalnoj grafici transformirat ćemo vrhove objekta koji su zadani kao jedno-redčane matrice  $1 \times 3$  ili  $1 \times 4$ . Matrice transformacija zadaju se kao matrice  $3 \times 3$  ili  $4 \times 4$ . Za to će nam biti potrebno množenje matrica. Također, transformacije mogu biti zadana kao niz matrica koje je potrebno množiti.

Kod inverznih transformacija ponekad se koriste i inverzne matrice te je potrebno imati i

ovu funkcionalnost za matrice  $3 \times 3$  ili  $4 \times 4$ .

### 1.3. Baricentrične koordinate

Baricentrične koordinate korisne su kod određivanja da li se točka nalazi u trokutu u 3D prostoru. Ako su vrhovi trokuta A, B, C. Za neku točku T imamo baricentričnu kombinaciju:

$$T = t_1 A + t_2 B + t_3 C$$

gdje su  $t_1, t_2, t_3$  baricentrične koordinate. Za baricentričnu kombinaciju vrijedi  $t_1 + t_2 + t_3 = 1$ .

Ovo možemo prikazati kao sustav jednadžbi, raspisan po koordinatama:

$$A_x t_1 + B_x t_2 + C_x t_3 = T_x$$

$$A_y t_1 + B_y t_2 + C_y t_3 = T_y$$

$$A_z t_1 + B_z t_2 + C_z t_3 = T_z$$

ili matrično:

$$\begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \text{ pa je rješenje } \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{bmatrix}^{-1} \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

Znači, da bi odredili baricentrične koordinate bit će potrebno invertirati matricu. Može se desiti da matrica nije invertibilna. To je na primjer slučaj kada je jedan vrh trokuta u ishodištu ili kada su točke trokuta u jednoj od ravnina xy, xz ili yz pa je jedna od koordinata nula. Tada je matrica singularna i ne možemo ju invertirati. U tom slučaju koristimo uvjet koji smo naveli da vrijedi za baricentrične koordinate  $t_1 + t_2 + t_3 = 1$  umjesto retka matrice koji je problematičan.

### 1.4. Radni zadatak

1. Napisati program koji izračunava i ispisuje na zaslonu:

$$\vec{v}_1 = (2\vec{i} + 3\vec{j} - 4\vec{k}) + (-\vec{i} + 4\vec{j} - \vec{k})$$

$$s = \vec{v}_1 \cdot (-\vec{i} + 4\vec{j} - \vec{k})$$

$$\vec{v}_2 = \vec{v}_1 \times (2\vec{i} + 2\vec{j} + 4\vec{k}), \text{ gdje je } \times \text{ vektorski produkt,}$$

$$\vec{v}_3 = |\vec{v}_2|, \text{ normirani vektor,}$$

$$\vec{v}_4 = -\vec{v}_2, \text{ vektor suprotnog smjera}$$

$$M_1 = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 4 & 5 & 1 \end{bmatrix} + \begin{bmatrix} -1 & 2 & -3 \\ 5 & -2 & 7 \\ -4 & -1 & 3 \end{bmatrix}$$

$$M_2 = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 4 & 5 & 1 \end{bmatrix} \begin{bmatrix} -1 & 2 & -3 \\ 5 & -2 & 7 \\ -4 & -1 & 3 \end{bmatrix}^T, \text{ gdje je } T \text{ oznaka za transponiranu matricu}$$

$$M_3 = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 4 & 5 & 1 \end{bmatrix} \begin{bmatrix} -1 & 2 & -3 \\ 5 & -2 & 7 \\ -4 & -1 & 3 \end{bmatrix}^{-1}$$

2. Napišite program koji će korisniku dopustiti da unese podatke o sustavu tri jednadžbe s tri nepoznanice (nije potrebno raditi parsiranje ulaza; pretpostavite da su varijable uvijek x, y i z, te da će korisnik unijeti redom podatke o prvoj jednadžbi pa podatke o drugoj jednadžbi te konačno podatke o trećoj jednadžbi). Program potom mora ispisati rješenje tog sustava. Primjerice, ako korisnik redom unese 1, 1, 1, 6, -1, -2, 1, -2, 2, 1, 3, 13, sustav jednadžbi koji je time definiran je:

$$1 \cdot x + 1 \cdot y + 1 \cdot z = 6$$

$$-1 \cdot x - 2 \cdot y + 1 \cdot z = -2$$

$$2 \cdot x + 1 \cdot y + 3 \cdot z = 13$$

a program treba ispisati da je rješenje  $[x \ y \ z] = [1 \ 2 \ 3]$

3. Napišite program koji će korisniku dopustiti da unese podatke o tri vrha trokuta (A, B i C) te dodatnoj točki T u 3D prostoru (potrebno je unijeti x, y i z koordinatu za svaki vrh odnosno točku). Program treba na zaslone ispisati baricentrične koordinate točke T s obzirom na zadani trokut.

Napomena: primijenite direktno prethodno opisani postupak i zanemarite moguće probleme uslijed loše odabranih vrhova odnosno zadane točke.

## 2. Pravac, linija

### 2.1. Homogene koordinate

Točka iz  $n$ -prostora može biti preslikana u homogenu točku u  $(n+1)$   $h$ -prostoru. Obrnuto, homogena točka iz  $(n+1)$   $h$ -prostora može biti projicirana u točku  $n$ -prostora. Promotrimo za primjer 2-prostor i njemu odgovarajući homogeni 3 $h$ -prostor.

Preslikavanje:

$$V(x\ y) \rightarrow V'(x'\ y'\ h),$$

pri tome je točka  $V$  u 2-prostoru a  $V'$  u 3 $h$ -prostoru i vrijedi:

$$\begin{aligned} x' &= hx, \\ y' &= hy. \end{aligned} \tag{2.1}$$

Projekcija:

$$V'(x'\ y'\ h) \rightarrow V(x\ y),$$

- pri tome vrijedi:

$$\begin{aligned} x &= x'/h, \\ y &= y'/h. \end{aligned} \tag{2.2}$$

Komponenta  $h$  zove se faktor proporcionalnosti ili homogena koordinata. Vrijednost homogene koordinate  $h$  je proizvoljna, najčešće se koristi slučaj  $h = 1$ . Ako je  $h = 0$  tada se radi o točki koja je u beskonačnosti u  $n$ -prostoru. Ako su pravci paralelni u  $n$ -prostoru tada su paralelni i u homogenom  $(n+1)$   $h$ -prostoru. Sačuvanost paralelnosti pravaca u homogenom prostoru važno je svojstvo.

Nešto o nazivu homogeni prostor. Jednadžba pravca u 2-prostoru glasi

$$ax+by+c=0, \tag{2.3}$$

ako se u 1.3 uvede izmjena 1.2 slijedi

$$\frac{ax'}{h} + \frac{by'}{h} + c = 0,$$

što daje

$$ax'+by'+ch=0. \tag{2.4}$$

Izraz 2.4 je jednadžba pravca u homogenom prostoru. Po svom obliku to je homogena jednadžba i zbog toga je i naziv homogeni prostor.

## 2.2. Jednadžba pravca

Pravac je određen s dvije točke, na primjer točke  $V_1$  i  $V_2$ . Koristi se homogeni prostor, tj.

$$V_1=(x_1 \ y_1 \ h_1), \ V_2=(x_2 \ y_2 \ h_2).$$

Vektorski oblik jednadžbe pravca određen je vektorskim produktom

$$P=V_1 \times V_2 = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_1 & y_1 & h_1 \\ x_2 & y_2 & h_2 \end{bmatrix} = \begin{bmatrix} y_1 h_2 - y_2 h_1 \\ -x_1 h_2 + x_2 h_1 \\ x_1 y_2 - x_2 y_1 \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (2.5)$$

Analitički oblik jednadžbe pravca, uz  $h_1=h_2=1$  je

$$\frac{y-y_1}{y_2-y_1} = \frac{x-x_1}{x_2-x_1}$$

odnosno

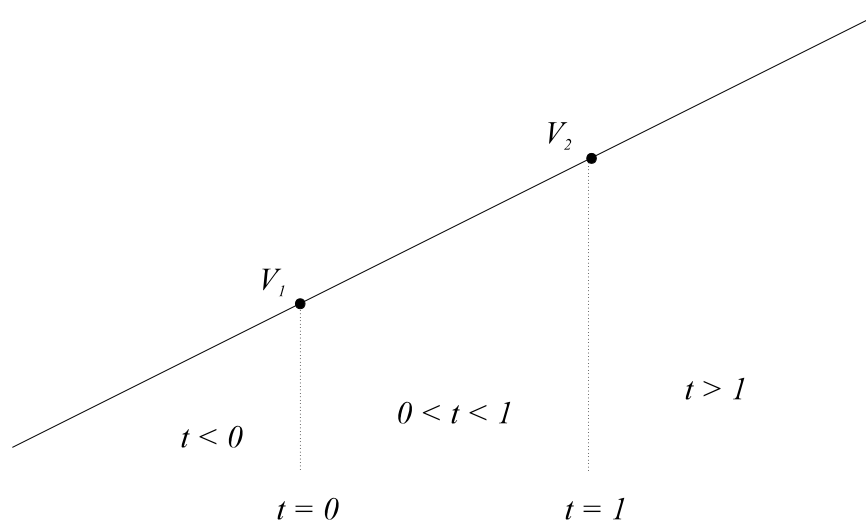
$$(y_1 - y_2)x + (-x_1 + x_2)y + x_1 y_2 - x_2 y_1 = ax + by + c = 0 \quad (2.6)$$

Parametarski oblik jednadžbe pravca je

$$P=(V_2-V_1)t+V_1, \text{ ili po koordinatama } \begin{aligned} x &= (x_2-x_1)t+x_1, \\ y &= (y_2-y_1)t+y_1, \\ h &= (h_2-h_1)t+h_1. \end{aligned} \quad (2.7)$$

Pri tome je točki  $V_1$  pridružena vrijednost parametra  $t=0$  a točki  $V_2$ , vrijednost parametra  $t=1$ .

Na slici 1.1. pokazano je pridjeljivanje vrijednosti parametra  $t$  dijelovima pravca, koji su od interesa.



Slika 2.1. Vrijednost parametra  $t$  i dijelovi pravca.



### 1.3. Ispitivanje odnosa točke i pravca

Skalarni produkt točke  $V(x \ y \ 1)$  i pravca  $P(a \ b \ c)^T$  određuje odnos točke i pravca, pri tome vrijedi dogovor:

$$V \cdot P = (x \ y \ 1) \begin{bmatrix} a \\ b \\ c \end{bmatrix} = a x + b y + c$$

$> 0$  točka  $V$  je iznad pravca  $P$   
 $= 0$  točka  $V$  je na pravcu  $P$   
 $< 0$  točka  $V$  je ispod pravca  $P$

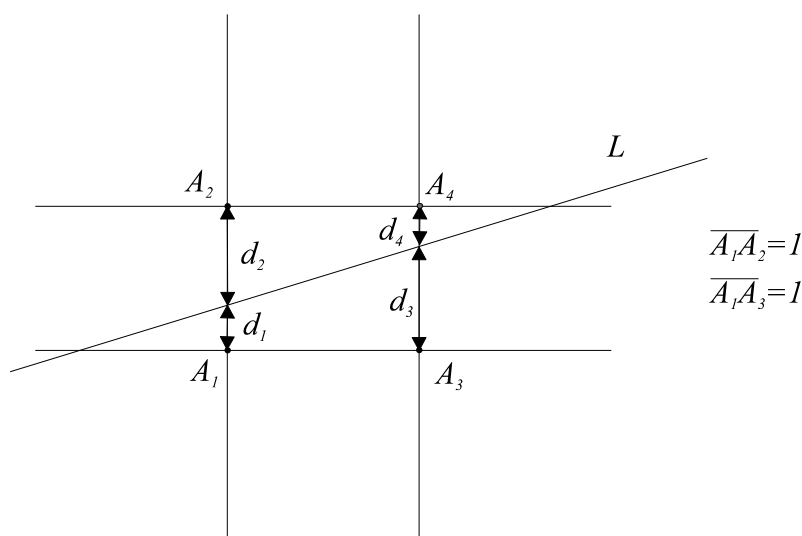
### 2.4. Iscrtavanje linije na zaslonu

Promatra se zaslon sličan TV ekranu (raster-scan). Između dvije točke zaslona potrebno je iscrtati ravnu liniju. U tu svrhu neophodan je postupak za izbor elemenata slike (pixel) koje treba osvijetliti. Ocjena valjanosti ovakvog postupka temelji se promatranjem:

- ravnosti linije,
- ispravnosti završetka linije,
- konstantnosti gustoće točaka u liniji i njena neovisnost o smjeru i duljini linije,
- brzine iscrtavanja.

#### 2.4.1. Bresenham-ov postupak

Za iscrtavanje linije na zaslonu sličnom TV ekranu najčešće se koristi Bresenham-ov postupak. Kriterij izbora točaka rastera sastoji se u računanju udaljenosti okolnih točaka rastera, od linije (slika 1.2.).



Slika 1.2. Bresenham-ov postupak.

Za liniju  $L$  treba osvijetliti točke  $A_1$  i  $A_4$  jer je  $d_1 < d_2$  i  $d_4 < d_3$ .

## 2.5. Radni zadatak

1. Učitati koordinate točkica  $V_1(x_1 \ y_1)$  i  $V_2(x_2 \ y_2)$ .
2. Koristeći Bresenham-ov postupak iscrtati liniju određenu točkama  $V_1$  i  $V_2$ .
3. Usporedba s *LINE* naredbom.  
Iscrtati liniju određenu koordinatama točkica  $(x_1 \ y_1+20)$  i  $(x_2 \ y_2+20)$  putem *LINE* naredbe.

## 2.6. Rješenje radnog zadatka

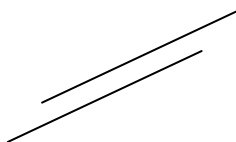
### 2.6.1. Postupak:

1. Učitati  $x$  y koordinate početne i završne točke linije  $(x_1 \ y_1), (x_2 \ y_2)$ .
2. Izračunati  $x_0, y_0$  intervale linije  $x_0=x_2-x_1, y_0=y_2-y_1$ .
3. Izračunati udaljenost  $D = y_0/x_0 - 0,5$ .
4. Postaviti koordinate tekuće točke  $x=x_1, y=y_1$ .
5. Za  $i = 0, i < x_0$  ponavljati korake 6-8. Ići na korak 9.
6. Osvijetliti tekuću točku  $x, y$ .
7. Za  $D > 0$  računati  $y=y+1, D=D-1$ .
8. Pribrojiti  $x=x+1, D=D+y_0/x_0$ .
9. Usporedba s *LINE* naredbom.  
Iscrtati liniju određenu koordinatama  $(x_1 \ y_1+20)$  i  $(x_2 \ y_2+20)$ .
10. Kraj.

### 1.6.2. Rezultati

Molim  $x, y$  koordinate početne točke ?    10    10

Molim  $x, y$  koordinate završne točke ?    200    100



Usporedba s *LINE* naredbom.

Opisani postupak radi ispravno za linije pod kutem  $0-45^\circ$ . Načiniti potrebne modifikacije koje će osigurati ispravno iscrtavanje linije pod bilo kojim kutem.

## 2.7. Za vježbu

Koje izmjene je potrebno načiniti u algoritmu da sve varijable postanu cjelobrojne?

### 3. Crtanje i popunjavanje konveksnog poligona

#### 3.1. Zadavanje poligona

Geometrijski i topološki podaci određuju poligon, slika 3.1. Geometrijski podaci su koordinate  $n$  vrhova poligona,

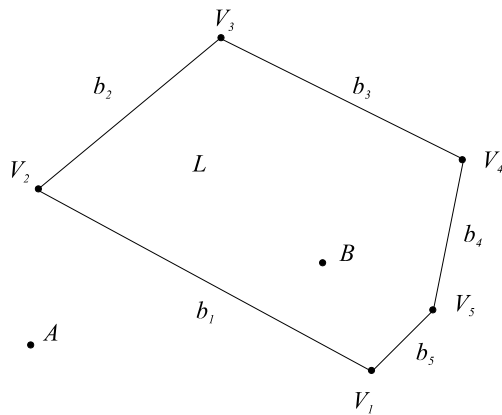
$$V_i = (x_i, y_i, h_i), \quad i = 1 \dots n.$$

Topološke podatke predstavlja popis vrhova poligona,

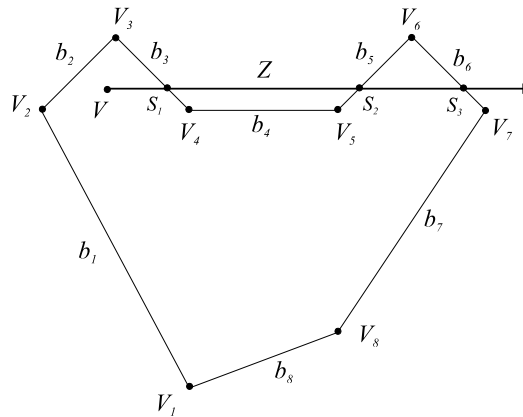
$$L = (V_i), \quad i = 1 \dots n.$$

Redoslijed vrhova u popisu  $L$  može biti u smjeru kazaljke na satu ili suprotan smjeru kazaljke na satu. Na primjer, za poligon na slici 3.1.a, to je

$$L = (V_1 V_2 V_3 V_4 V_5) \text{ ili } L = (V_1 V_5 V_4 V_3 V_2).$$



a) Konveksan poligon.



b) Konkavan poligon.

Slika 3.1. Poligoni.

Jednadžba pravca u kome leži brid  $b_i$  poligona  $L$ , ili kraće jednadžba brida  $b$ , određena je vektorskim produktom početnog i završnog vrha brida,

$$\begin{aligned} b_i &= V_i \times V_{i+1}, & i &= 1 \dots n-1, \\ b_n &= V_n \times V_1, & i &= n. \end{aligned} \quad (3.1)$$

#### 3.2. Provjera orijentacije bridova

Redoslijed vrhova u popisu može biti zahtijevan kao ulazni podatak, ali i ne mora. Uz pretpostavku da je redoslijed vrhova u popisu u smjeru kazaljke na satu, za konveksan poligon vrijedi kriterij:

$$(\forall i) (V_j b_i < 0), \quad \begin{aligned} j &= i+2 \text{ za } i \leq n-2, \\ j &= i+2-n \text{ za } i > n-2. \end{aligned} \quad i = 1 \dots n, \quad (3.2)$$

Ako se traži redoslijed vrhova u popisu u smjeru kazaljke na satu, a kriterij 3.2. nije

ispunjen potrebne su ispravke:

- okrenuti redoslijed vrhova u popisu,
- ponoviti računanje koeficijenata jednadžbi bridova.

### 3.3. Ispitivanje odnosa točke i poligona

Za konveksan poligon provjeru orijentacije bridova možemo načiniti na slijedeći način. Točka  $A$  je izvan poligona  $L$ , slika 3.1.a, jer zadovoljava kriterij

$$(\exists i)(Ab_i > 0), i = 1..n. \quad (3.3)$$

Točka  $B$  je unutar poligona  $L$  jer zadovoljava kriterij

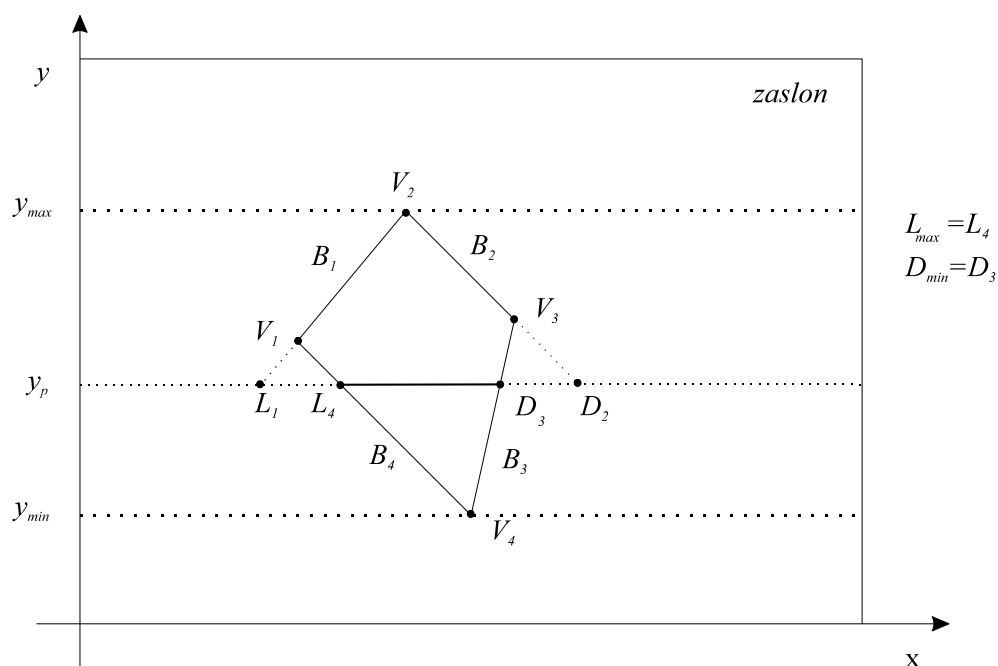
$$(\forall i)(Bb_i < 0), i = 1..n. \quad (3.4)$$

### 3.4. Bojanje konveksnog poligona

Brid poligona  $b_i$  određen je početnim vrhom  $V_i$  i završnim vrhom  $V_{i+1}$ . Bridove poligona treba razvrstati na "lijeve" i "desne" bridove po načelu:

- ako vrijedi  $y_i < y_{i+1}$  brid je lijevi,
- ako vrijedi  $y_i > y_{i+1}$  brid je desni.

Odrediti sva sjecišta lijevih i desnih bridova s linijom prikaza  $y_p$ , slika 3.2. Pronaći  $L_{max}$ , sjecište lijevih bridova s najvećom  $x$  koordinatom. Pronaći  $D_{min}$ , sjecište desnih bridova s najmanjom  $x$  koordinatom. Obojiti dio linije prikaza  $L_{max}$ ,  $D_{min}$ . Postupak ponoviti za sve linije prikaza. U slučaju  $L_{max} > D_{min}$  poligon je ispod ili iznad linije prikaza  $y_p$ . Područje odabira linije prikaza možemo ograničiti na područje poligona tj. između  $y_{min}$  i  $y_{max}$ .



Slika 3.2. Bojanje konveksnog poligona.

### 3.5. Radni zadatak

1. Zadati koordinate  $n$  vrhova konveksnog poligona, uz redoslijed vrhova u smjeru kazaljke na satu.
2. Iscrtati poligon na zaslonu.
3. Izračunati koeficijente jednadžbi bridova.
4. Zadati koordinate točke  $V$  i ispitati odnos točke  $V$  i poligona.
5. Obojati poligon.

### 3.6. Rješenje radnog zadatka

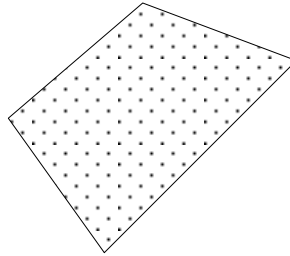
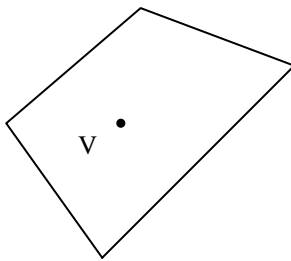
1. Učitati broj vrhova  $n$  konveksnog poligona.
2. Učitati  $x$   $y$  koordinate vrhova,  $x(i)$ ,  $y(i)$ ,  $i = 0, n-1$ . Redoslijed vrhova neka je u smjeru kazaljke na satu. Odrediti  $y_{min}$ ,  $y_{max}$ , i  $x_{min}$ ,  $x_{max}$ . (unos točaka moguće je napraviti i interaktivno upotrebom miša).
3. Postaviti  $x(n)=x(0)$ ,  $y(n)=y(0)$ ,
4. Iscrtati poligon.
5. Izračunati koeficijente jednadžbi bridova.

$$\begin{aligned} a(i) &= y(i) - y(i+1), \\ b(i) &= -x(i) + x(i+1), \\ c(i) &= x(i) \cdot y(i+1) - x(i+1) \cdot y(i). \end{aligned} \quad i = 0, n-1$$

6. Učitati  $x$   $y$  koordinate točke  $V(x_1, y_1)$ .
7. Ispitati odnos točke  $V$  i poligona.  
Ako postoji barem jedan brid tako da vrijedi  
 $x_1 a(i) + y_1 b(i) + c(i) > 0$ ,  $i = 0, n-1$  točka  $V$  je izvan poligona,  
inače, točka  $V$  je unutar poligona.
8. Bojanje poligona.  
Za sve ispitne linije  $Y_0 = y_{min}$ ,  $y_{max}$  izvesti korake 9-15. Ići na korak 16.
9. Postaviti  $L = x_{min}$ ,  $D = x_{max}$ .
10. Za  $i = 0, n-1$  ponavljati korake 11-14. Ići na korak 15.
11. Ako je  $A(i)=0$  ne izvoditi korake 12 -14.
12. Računati  $x$  koordinatu sjecišta ispitne linije  $y_0$  i  $i$ -tog brida,  
 $x_i = [-b(i) y_0 - c(i)] / a(i)$ .
13. Lijevi brid.  
Ako je  $y(i) < y(i+1)$  tada ako je  $x_i > L$  postaviti  $L = x_i$ .
14. Desni brid.  
Ako je  $y(i) \geq y(i+1)$  tada ako je  $x_i < D$  postaviti  $D = x_i$ .
15. Ako je  $L < D$  iscrtati liniju  $(L, y_0)$ ,  $(D, y_0)$ .
16. Kraj.

Rezultati

Molim broj vrhova poligona?	4	
Molim koordinate vrha?	50	200
Molim koordinate vrha?	150	350
Molim koordinate vrha?	300	150
Molim koordinate vrha?	100	50
Molim koordinate točke?	150	200



TOČKA V JE UNUTAR POLIGONA !

TOČKA V JE IZVAN POLIGONA !

## 4. Ravnina, tijelo

### 4.1. Jednadžba ravnine

Ravnina je određena s tri nekolinearne točke

$$\begin{aligned} V_1 &= (x_1 \quad y_1 \quad z_1 \quad h_1), \\ V_2 &= (x_2 \quad y_2 \quad z_2 \quad h_2), \\ V_3 &= (x_3 \quad y_3 \quad z_3 \quad h_3). \end{aligned}$$

Dva su osnovna oblika jednadžbe ravnine: analitički oblik i parametarski oblik jednadžbe ravnine.

### 4.2. Analitički oblik jednadžbe ravnine

Analitički oblik jednadžbe ravnine je:

$$Ax + By + Cz + D = 0.$$

(uz  $h_1 = h_2 = h_3 = 1$ ) vrijedi

$$\text{determinanta} \begin{vmatrix} x - x_1 & y - y_1 & z - z_1 \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \end{vmatrix} = Ax + By + Cz + D = 0. \quad (4.1)$$

Koeficijenti jednadžbe ravnine, u slučaju kada  $n$  točaka točno ili približno leži u ravnini, određeni su postupkom

$$\begin{aligned} A &= \sum_{i=0}^{n-1} (y_i - y_j)(z_i + z_j) \\ B &= \sum_{i=0}^{n-1} (z_i - z_j)(x_i + x_j) \quad \begin{matrix} j = i + 1 & \text{za } i \neq n \\ j = 0 & \text{za } i = n - 1. \end{matrix} \\ C &= \sum_{i=0}^{n-1} (x_i - x_j)(y_i + y_j) \end{aligned} \quad (4.2)$$

Za  $i$ -tu točku s koordinatama  $(x_i, y_i, z_i)$  koja leži u ravnini vrijedi:

$$Ax_i + By_i + Cz_i + D = 0,$$

što daje

$$D = -Ax_i - By_i - Cz_i.$$

Jednadžbu ravnine  $R$  možemo odrediti na osnovi tri točke, odnosno na osnovi tri skalarna produkta

$$V_1 R = 0, \quad V_2 R = 0, \quad V_3 R = 0,$$

odnosno u matričnom obliku:

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} R = \begin{bmatrix} 0 \\ 0 \\ 0 \\ D \end{bmatrix}$$

iz čega slijedi

$$R = \begin{bmatrix} x_1 & y_1 & z_1 & h_1 \\ x_2 & y_2 & z_2 & h_2 \\ x_3 & y_3 & z_3 & h_3 \\ 0 & 0 & 0 & D \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} \quad (4.3)$$

gdje su  $A, B, C$  koeficijenti vektora normale na ravninu. U slučaju kada su tri promatrane točke kojima želimo odrediti ravninu kolinearne (leže na istom pravcu) rješenje nije jednoznačno. Ako je ravnina određena zadanim točkama koplanarna s ravninama  $xy, xz, yz$ , bit će potrebno posebno razmatranje i određivanje vektora normale ravnine  $(0 \ 0 \ 1), (0 \ 1 \ 0), (1 \ 0 \ 0)$ .

Koeficijente  $A, B, C$  vektora normale na ravninu možemo odrediti i kao vektorski produkt dvaju vektora  $(V_2 - V_1)$  i vektora  $(V_3 - V_1)$ :

$$\vec{n} = (A \ B \ C) = (V_2 - V_1) \times (V_3 - V_1) \quad (4.4)$$

gdje redoslijed vrhova određuje smjer vektora normale ravnine (prema «gore» ili «dolje»). Ova formula odgovara formuli (4.1).

### 4.3. Parametarski oblik jednadžbe ravnine

Parametarski oblik jednadžbe ravnine po koordinatama linearna je funkcija dvaju parametara  $u$  i  $w$ .

$$\begin{aligned} x &= a_1 u + b_1 w + c_1, \\ y &= a_2 u + b_2 w + c_2, \\ z &= a_3 u + b_3 w + c_3, \\ h &= a_4 u + b_4 w + c_4, \end{aligned} \quad (4.5)$$

Iz 4.5 slijedi matrični oblik

$$V = \begin{bmatrix} x & y & z & h \end{bmatrix} = \begin{bmatrix} u & w & 1 \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \end{bmatrix}, \quad (4.6)$$

odnosno

$$V = \begin{bmatrix} x & y & z & h \end{bmatrix} = \begin{bmatrix} u & w & 1 \end{bmatrix} K, \quad (4.7)$$

gdje je  $K$  karakteristična matrica ravnine.

Ako se točkama  $V_1, V_2$  i  $V_3$  pridijeli vrijednost parametara

$$\begin{aligned} u &= 0, w = 0 \text{ za } V_1, \\ u &= 1, w = 0 \text{ za } V_2, \\ u &= 0, w = 1 \text{ za } V_3. \end{aligned}$$



iz 4.6 i 4.7 slijedi

$$K = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x_1 & y_1 & z_1 & h_1 \\ x_2 & y_2 & z_2 & h_2 \\ x_3 & y_3 & z_3 & h_3 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \end{bmatrix} \quad (4.8)$$

#### 4.4. Ispitivanje odnosa točke i ravnine

Skalarni produkt točke i ravnine (slično kao kod pravca i točke) određuje odnos točke i ravnine, pri tome po dogovoru vrijedi:

$$V R = Ax + By + Cz + D \begin{cases} > 0 & \text{točka je iznad ravnine,} \\ = 0 & \text{točka u ravnini,} \\ < 0 & \text{točka je ispod ravnine,} \end{cases} \quad (4.9)$$

#### 4.5. Probodište pravca i ravnine

Probodište je zajednička točka  $V$  pravca  $P$  i ravnine  $R$ ,

$$V \cdot P = 0 \quad \text{i} \quad V \cdot R = 0,$$

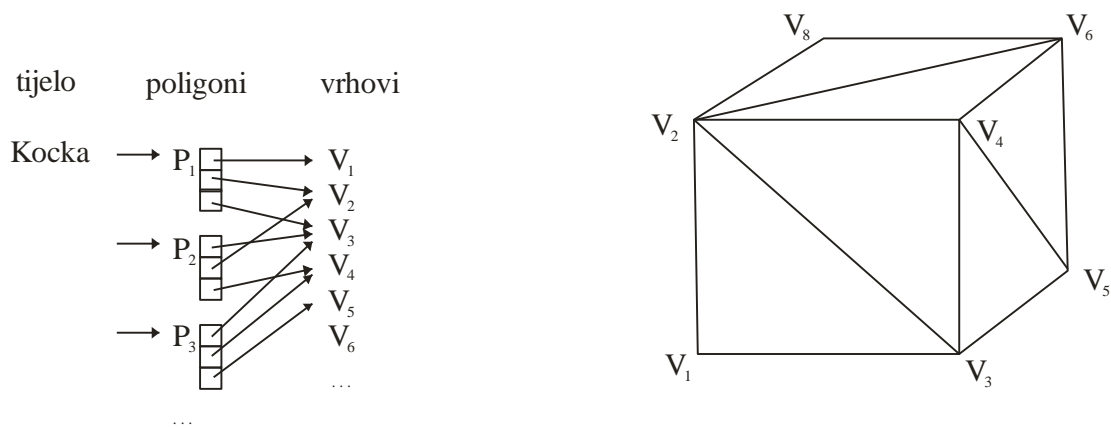
Na osnovi ova dva uvjeta možemo naći točku u kojoj parametarski zadan pravac probada analitički zadanu ravninu. Odredit ćemo parametar  $t$  za koji vrijedi da je točka pravca istovremeno i točka ravnine.

#### 4.6. Zadavanje tijela

Tijelo je zadano geometrijskim i topološkim podacima. Geometrijski podaci su vrhovi, odnosno njihove koordinate. Topološke podatke čini povezanost vrhova, odnosno popisi koji određuju površinu, poligone i bridove, te pripadne vrhove. U ovom primjeru odabrana je organizacija prikazana na slici 4.1. Ovo je jedan od načina organizacije topoloških i geometrijskih podataka, no ovisno o daljnjim postupcima koji će biti načinjeni nad objektom, mogu se koristiti i drugačiji oblici pohranjivanja strukture podataka o objektu.

Trokuti su najčešće korišteni poligoni u opisu tijela zato što je ravnina određena s tri točke. Na primjer, kod tijela koje je zadano četverokutima, sve četiri točke ne leže nužno u ravnini, što može izazvati probleme.

Važno je strukturu podataka načiniti tako da je poznat redoslijed vrhova za pojedini poligon. Redoslijed vrhova možemo odabrati tako da bude u smjeru suprotno kazaljke na satu gledano izvan tijela. Redoslijed vrhova određuje orijentaciju normala poligona. Za neko tijelo normale svih poligona moraju biti usmjerene na isti način, na primjer sve u unutrašnjost tijela ili sve izvan tijela.



Slika 4.1. Tijelo i pripadna struktura podataka.

Tijelo se može sastojati i od površina različite boje. U tom slučaju se poligoni grupiraju u skupine istih svojstava.

Za primjer na slici 4.1. popis vrhova, tako da je redoslijed vrhova poligona u smjeru suprotno kazaljke na satu gledano izvan tijela, izgledao bi ovako:

Kocka:		Podaci u datoteci za vrhove i poligone:					
P <sub>1</sub>	V <sub>0</sub> V <sub>1</sub> V <sub>2</sub>	v	0.0	0.0	0.0	f	1 3 2
P <sub>2</sub>	V <sub>2</sub> V <sub>1</sub> V <sub>3</sub>	v	0.0	0.0	1.0	f	3 4 2
P <sub>3</sub>	V <sub>2</sub> V <sub>3</sub> V <sub>4</sub>	v	1.0	0.0	0.0	f	3 5 4
P <sub>4</sub>	V <sub>4</sub> V <sub>3</sub> V <sub>5</sub>	v	1.0	0.0	1.0	f	5 6 4
P <sub>5</sub>	V <sub>4</sub> V <sub>5</sub> V <sub>6</sub>	v	1.0	1.0	0.0	f	5 7 6
P <sub>6</sub>	V <sub>6</sub> V <sub>5</sub> V <sub>7</sub>	v	1.0	1.0	1.0	f	7 8 6
P <sub>7</sub>	V <sub>6</sub> V <sub>7</sub> V <sub>0</sub>	v	0.0	1.0	0.0	f	7 1 8
P <sub>8</sub>	V <sub>0</sub> V <sub>7</sub> V <sub>1</sub>	v	0.0	1.0	1.0	f	1 2 8
P <sub>9</sub>	V <sub>0</sub> V <sub>2</sub> V <sub>4</sub>	#	komentar			f	1 5 3
P <sub>10</sub>	V <sub>0</sub> V <sub>4</sub> V <sub>7</sub>					f	1 7 5
P <sub>11</sub>	V <sub>1</sub> V <sub>5</sub> V <sub>3</sub>					f	2 4 6
P <sub>12</sub>	V <sub>1</sub> V <sub>7</sub> V <sub>5</sub>					f	2 6 8

#### 4.7. Orijentacija normale ravnine

Vektor normale  $\mathbf{n}$  ravnine  $R$  može biti usmjeren

- u unutrašnjost tijela,
- izvan tijela.

Tri susjedna nekolinearna vrha  $V_{i-1}$ ,  $V_i$ ,  $V_{i+1}$  u popisu vrhova poligona određuju koeficijente jednadžbe ravnine  $R$ ,

$$Ax + By + Cz + D = 0, \quad (4.10)$$

a time i komponente vektora normale  $\mathbf{n}$  na ravninu  $R$

$$\mathbf{n} = [A \quad B \quad C]. \quad (4.11)$$

Uz pretpostavke:

- redoslijed vrhova  $V_{i-1}$ ,  $V_i$ ,  $V_{i+1}$  je u smjeru suprotno kazaljke na satu gledano izvan tijela
  - srednji vrh je na konveksnom dijelu poligona (za trokute uvijek ispunjeno)
- vektor normale  $\mathbf{n}$  ravnine  $R$  usmjeren je izvan tijela.

#### 4.8. Ispitivanje odnosa točke i konveksnog tijela

U slučaju kada se radi o konveksnom tijelu, te kada je vektor normale ravnine usmjeren izvan tijela, točka  $V$  je unutar tijela ako vrijedi

$$(\forall i)(VR_i < 0), i = 1..n, \quad (4.12)$$

odnosno, točka  $V$  je izvan tijela ako vrijedi

$$(\exists i)(VR_i > 0), i = 1..n. \quad (4.13)$$

Ako je tijelo konkavno tada se iz točke povlači zraka i određuje probodište zrake i stranice tijela. Ako je broj probodišta

- paran, točka je izvan tijela,
- neparan, točka je unutar tijela.

#### 4.9. Zapisi poligonalnih objekata

Zapisivanje trodimenzijskih objekata obično je propisano specifikacijama proizvođača programske opreme. To su, na primjer, zapisi s ekstenzijom .3ds, .max, .dxf, .ply, .obj, .xgl, .stl, .wrl, .iges ... . Ovisno o zapisu, zapisani mogu biti trokuti ili poligoni s više vrhova, grupe poligona, boje, teksture, normale u poligonima ili vrhovima, krivulje, površine, način konstrukcije objekata, podaci o sceni, izvori svjetla, podaci o animaciji i slično. U vježbi će biti korišten zapis .obj, odnosno dio tog zapisa.

#### 4.10. Radni zadatak

Zadan je jednostavan primjer u kojem se koristi samo dio zapisa objekta Wavefront (.obj). Zapis sadrži popis vrhova i njihovih koordinata te popis poligona s pripadnim indeksima vrhova.

1. Iz datoteke učitati zadano tijelo (za primjer sa slike 4.1)
  - u prvom prolazu izbrojiti broj vrhova i poligona (prvo slovo  $v$ ,  $f$  ili  $\#$ )
  - načiniti alokaciju potrebnog memorijskog prostora
  - učitati geometrijske i topološke podatke (vrhove i poligone)
  - zadati koordinate ispitne točke  $V$ .
2. Odrediti  $A_i$ ,  $B_i$ ,  $C_i$ ,  $D_i$  koeficijente jednadžbe ravnine, za svaki poligon tijela. Koristiti formulu 4.1. odnosno:
 
$$A = (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1)$$

$$B = -(x_2 - x_1)(z_3 - z_1) + (z_2 - z_1)(x_3 - x_1)$$

$$C = (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)$$

$$D = -x_1A - y_1B - z_1C$$
3. Za ispitnu točku  $V$  odrediti da li je unutar ili izvan konveksnog tijela. Koristiti uvjete 4.12, 4.13. Postaviti  $z$  koordinatu  $z=0$  i iscrtati tijelo (po potrebi translirati i skalirati dobivenu sliku).

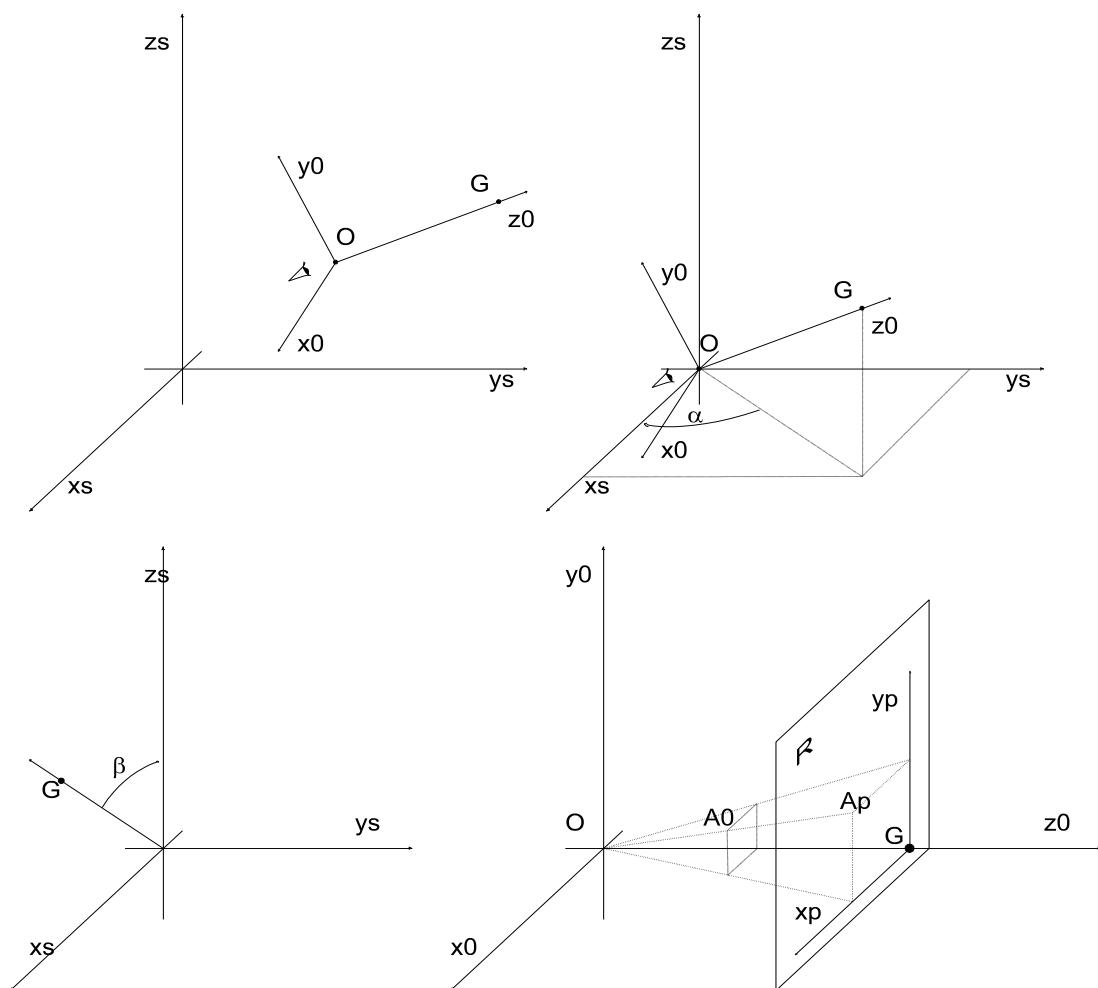
## 5. Transformacija pogleda i perspektivna projekcija

Sustav scene je trodimenzijski sustav, slika 5.1. Sustav oka je trodimenzijski sustav čija je  $z$  os upravljena u smjeru pogleda, tako da  $z$  os predstavlja dubinu slike. Sustav prikaza je dvodimenzijski sustav i smješten je u ravnini projekcije  $R$ .

Preslikavanje točaka iz sustava scene u sustav oka naziva se transformacija pogleda. Projekcija točaka iz sustava oka u sustav prikaza može se načiniti kao paralelna ili perspektivna projekcija.

Korištene oznake:

- $x_S y_S z_S$  - sustav scene,
- $x_O y_O z_O$  - sustav oka,
- $x_P y_P$  - sustav prikaza,
- $O$  - očište, položaj promatrača u sustavu scene,
- $G$  - gledište, točka u sustavu scene u koju je usmjeren pogled,
- $H$  - udaljenost ravnine projekcije od očišta,  $H=d(O,G)$ ,
- $R$  - ravnina projekcije, točka  $G$  leži u ravnini projekcije,



Slika 5.1. Sustav scene, sustav oka i sustav prikaza.

### 5.1. Transformacija pogleda

Za transformaciju pogleda treba odrediti matricu  $T$  tako da vrijedi

$$A_o = A_s T \quad (5.1)$$

Matrica  $T$  je sastavljena matrica od pet matrica elementarnih transformacija, to su:

$T_1$  - pomak ishodišta u točku  $O$ ,

$T_2$  - rotacija za kut  $\alpha$  oko  $z$  osi,

$T_3$  - rotacija za kut  $\beta$  oko  $y$  osi,

$T_4$  - rotacija za kut  $90^\circ$  oko  $z$  osi,

$T_5$  - promjena predznaka na  $x$  osi.

Točke  $O(x_o \ y_o \ z_o)$  i  $G(x_g \ y_g \ z_g)$  mjere se u sustavu scene. Koordinatama točke  $O$  određena je matrica  $T_1$

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_o & -y_o & -z_o & 1 \end{bmatrix}. \quad (5.2)$$

Djelovanje  $T_1$  na  $G$  daje

$$\begin{aligned} x_{g1} &= x_g - x_o \\ G_1 = GT_1 \text{ ili } y_{g1} &= y_g - y_o \\ z_{g1} &= z_g - z_o \end{aligned}$$

Matrica  $T_2$  glasi

$$T_2 = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5.3)$$

pri čemu je

$$\sin \alpha = \frac{y_{g1}}{\sqrt{x_{g1}^2 + y_{g1}^2}}, \quad \cos \alpha = \frac{x_{g1}}{\sqrt{x_{g1}^2 + y_{g1}^2}}$$

Djelovanje matrice  $T_2$  na  $G_1$  daje

$$\begin{aligned} x_{g2} &= \sqrt{x_{g1}^2 + y_{g1}^2} \\ G_2 = G_1 T_2 \text{ ili } y_{g2} &= 0 \\ z_{g2} &= z_{g1} \end{aligned}.$$

Matrica  $T_3$  glasi

$$T_3 = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5.4)$$

pri čemu je

$$\sin\beta = \frac{x_{g2}}{\sqrt{x_{g2}^2 + z_{g2}^2}}, \quad \cos\beta = \frac{z_{g2}}{\sqrt{x_{g2}^2 + z_{g2}^2}}.$$

Djelovanje matrice  $T_3$  na  $G_2$  daje

$$G_3 = G_2 T_3 \quad \text{ili} \quad \begin{aligned} x_{g3} &= 0 \\ y_{g3} &= 0 \\ z_{g3} &= \sqrt{x_{g2}^2 + z_{g2}^2} \end{aligned}$$

Nakon ovih transformacija ostvareno je podudaranje  $z$  osi koordinatnog sustava oka  $z_o$  sa  $z$  osi koordinatnog sustava scene  $z_s$ . Potrebno je još ostvariti podudaranja  $x$  i  $y$  osi sustava oko sa sustavom scene. Koordinatni sustav na zaslonu obično je postavljen tako da je ishodište u gornjem lijevom uglu zaslona,  $x$  os postavljena je horizontalno prema desno  $y$  os vertikalno prema dolje a  $z$  os usmjerena je prema promatraču. Koordinatni sustav oka je prema tome lijevi koordinatni sustav, a sustav scene desni. Potrebne matrice  $T_4$  i  $T_5$  za ostvarivanje podudaranja  $x$  i  $y$  osi sustava oka sa sustavom scene glase

$$T_4 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad T_5 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.5)$$

Matrica  $T$  je umnožak

$$T = T_1 T_2 T_3 T_4 T_5. \quad (5.6)$$

## 5.2. Perspektivna projekcija

Zadaća je odrediti matricu  $P$  koja će po zakonu perspektive projicirati točke iz sustava oka u ravninu projekcije, slika 5.1, odnosno u sustavu prikaza,

$$A_p = A_0 P. \quad (5.7)$$

Udaljenost ravnine projekcije od očišta je

$$H = \sqrt{(x_o - x_g)^2 + (y_o - y_g)^2 + (z_o - z_g)^2} = z_{g3}. \quad (5.8)$$

Iz sličnosti trokuta slijedi

$$x_p = \frac{x_o}{z_o} H, \quad y_p = \frac{y_o}{z_o} H. \quad (5.9)$$

Izraz 5.9 napisan u matričnom obliku glasi

$$A_p' = A_0 P$$

ili po koordinatama

$$\begin{bmatrix} x_p' & y_p' & z_p' & h_p' \end{bmatrix} = \begin{bmatrix} x_0 & y_0 & z_0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{H} \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\begin{aligned} x_p' &= x_0 \\ y_p' &= y_0 \\ z_p' &= 0 \\ h_p' &= \frac{z_0}{H} \end{aligned} \tag{5.10}$$

Iz 5.10 slijedi 5.9, što je povratak u nehomogeni prostor tj.

$$x_p = \frac{x_p'}{h_p'} = \frac{x_0}{z_0} H, \quad y_p = \frac{y_p'}{h_p'} = \frac{y_0}{z_0} H.$$

### 5.3. Radni zadatak

Zadati poligon te načiniti transformaciju pogleda i perspektivnu projekciju.

1. Iz datoteke učitati koordinate očišta, gledišta i vrhova poligona u sustavu scene. Gledište se obično zadaje u ishodištu scene  $G = (0 \ 0 \ 0)$  ili u središtu tijela (poligona). Očište je točka iz koje gledamo i ovisit će o veličini objekta. Ako su koordinate objekta u rasponu  $(-1, 1)$  očište može biti npr.  $O = (1 \ 1 \ 3)$ . Moramo paziti da očište ne zadamo u unutrašnjosti objekta.
2. Odrediti matricu transformacije pogleda  $T$  po formuli 5.6.
3. Odrediti matricu perspektivne projekcije  $P$ .
4. Načiniti transformaciju i projekciju zadanih vrhova poligona.
5. Iscrtati poligon.
6. Ponoviti korake 1-5. za tijelo iz prethodne vježbe.

## 6. Prikaz prostornih krivulja postupkom Beziera

Bezierove krivulje možemo definirati na dva načina. U slučaju korištenja gibanja vrha otvorenog sastavljenog poligona krivulju definiramo Bezierovim težinskim funkcijama. Drugi slučaj je kada su težinske funkcije vezane uz vrhove kontrolnog poligona i definirane su Bernstein-ovim polinomima.

### 6.1. Bezierove krivulje definirane Bezierovim težinskim funkcijama

Krivulja Beziera definirana Bezierovim težinskim funkcijama je:

$$\mathbf{p}(t) = \sum_{i=0}^n \mathbf{a}_i f_{i,n}(t) \quad (1)$$

gdje su:

$\mathbf{p}(t)$     točke na krivulji (vektorski zapis  $x(t)$ ,  $y(t)$ ,  $z(t)$ ),  
 $\mathbf{a}_i$         vektori kontrolnog poligona (između točke koje zadajemo),  
 $f_{i,n}$         bazne (težinske) funkcije stupnja  $n$ .

Bazne funkcije definirane su polinomima:

$$f_{i,n} = \frac{(-t)^i}{(i-1)!} \frac{d^{i-1} \Phi_n(t)}{dt^{i-1}}, \quad \Phi_n(t) = \frac{1-(1-t)^n}{-t} \quad i = 1 \dots n \quad (2)$$

ili rekurzivnom formulom:

$$f_{i,n}(t) = (1-t)f_{i,n-1}(t) + t f_{i-1,n-1}(t), \quad f_{0,0}(t) = 1, f_{k+1,k}(t) = 0, f_{-1,k}(t) = 1 \quad (3)$$

### 6.2. Bezierove krivulje definirane polinomima Bernsteina

Krivulja Beziera definirana Bernsteinovim težinskim funkcijama je:

$$\mathbf{p}(t) = \sum_{i=0}^n \mathbf{r}_i b_{i,n}(t) \quad (4)$$

gdje su:

$\mathbf{p}(t)$     točke na krivulji (vektorski zapis  $x(t)$ ,  $y(t)$ ,  $z(t)$ ),  
 $\mathbf{r}_i$         vrhovi kontrolnog poligona (točke koje zadajemo),  
 $b_{i,n}$         bazne (težinske) funkcije stupnja  $n$ .

Bazne funkcije definirane su Bernsteinovim polinomima:

$$b_{i,n} = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} \quad (5)$$



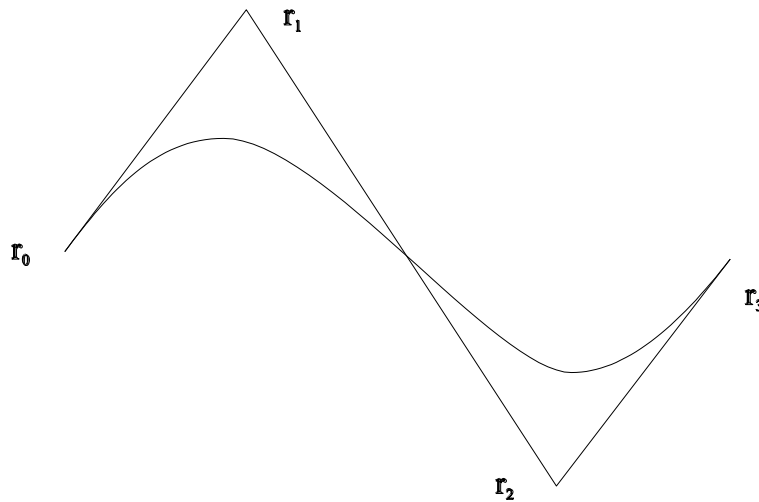
Primjer težinskih funkcija za četiri kontrolne točke (kubna krivulju):

$$b_{0,3} = (1-t)^3$$

$$b_{1,3} = 3t(1-t)^2$$

$$b_{2,3} = 3t^2(1-t)$$

$$b_{3,3} = t^3$$



Slika 1. Primjer krivulje zadane sa četiri točke.

### 6.3. Radni zadatak:

1. Učitati iz datoteke  $n+1$  točku kontrolnog poligona.
2. Iscrtati poligon.
3. Mijenjati parametar  $t$  od nule do jedan s korakom 0,01.
4. Prema formuli (4) i (5) za svaki  $t$  odrediti koordinate  $x(t)$ ,  $y(t)$  i  $z(t)$  krivulje, te iscrtati točke.
5. Dobiveni postupak upotrijebiti za određivanje putanje očišta (ili gledišta ili tijela) u prethodnoj vježbi. Iscrtavati žičnu formu tijela uz uklanjanje stražnjih poligona. Uklanjanje stražnjih poligona ostvariti na osnovi provjere kuta između vektora normale i vektora prema promatraču (poglavlje 8.2 iz knjige <http://www.zemris.fer.hr/predmeti/irg/knjiga.pdf>).

## 7. Sjenčanje tijela

U postupak sjenčanja tijela uključeni su postupci iz prethodnih poglavlja. To je postupak izgradnje tijela iz poglavlja 4, postupak transformacije pogleda i projekcije iz poglavlja 5. U ovoj vježbi će za transformaciju pogleda biti korištena funkcija OpenGL-a ili (Direct3D), kako bi ju mogli usporediti s načinjenim postupkom iz 5 vježbe. Cilj vježbe je načiniti prikaz tijela bez sjenčanja uz konstantno sjenčanje i uz Gouraud-ovo (izgovara se Guró) sjenčanje, te načiniti uklanjanje stražnjih poligona.

Za konstantno sjenčanje potrebno je poznavati normale poligona. Za Gouraud-ovo sjenčanje potrebno je odrediti normale u vrhovima objekta. Na osnovi normala određuje se intenzitet i boji se poligon. Iscrtavanje objekata poželjno ubrzati tako da uklonimo stražnje poligone.

### 7.1. Uklanjanje skrivenih linija i površina

Postoje različiti postupci za uklanjanje skrivenih linija i površina. To su postupci temeljeni na geometrijskim izračunavanjima u prostoru scene ili ravnine projekcije. Osnovni zahvati koriste mnoga izračunavanja i vremenski su zahtjevni. U svrhu poboljšanja efikasnosti postupaka, nastoje se primijeniti brzi zahvati koji će smanjiti trajanje osnovnog zahvata. U brze zahvate koje možemo koristiti ubrajamo

- uklanjanje stražnjih poligona,
- minimaks provjere.

Uklanjanje stražnjih poligona

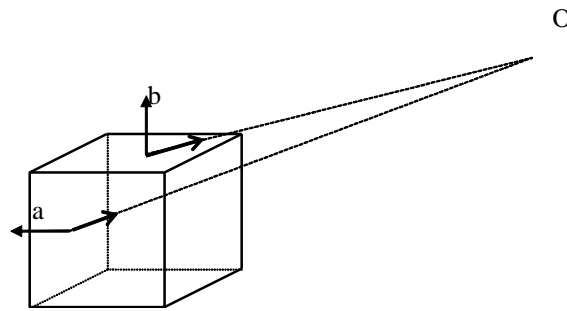
Uklanjanje stražnjih poligona možemo načiniti promatranjem u sustavu scene ili u sustavu prikaza.

Uklanjanje stražnjih poligona u sustavu scene možemo načiniti na osnovi promatranja kuta između vektora normale ravnine koju određuje poligon i vektora od promatranog poligona (središnje točke poligona) do očista *Slika 7.1*. Ako su vektori normala svih poligona usmjereni u unutrašnjost tijela, kut između promatranih vektora određuje da li je poligon stražnji

- $0^\circ$ -  $90^\circ$  ako je poligon nije stražnji,
- $90^\circ$ - $180^\circ$  ako poligon stražnji.

Možemo primijetiti da nije neophodno računati kut između normale ravnine i vektora prema očistu. Dovoljno je ispitati s koje strane ravnine (u kojoj leži promatrani poligon) leži točka očista. Odnosno, ako je produkt točke očista i jednadžbe ravnine poligona (4.9):

- pozitivan, poligon je vidljiv,
- negativan ili nula, poligon nije vidljiv.



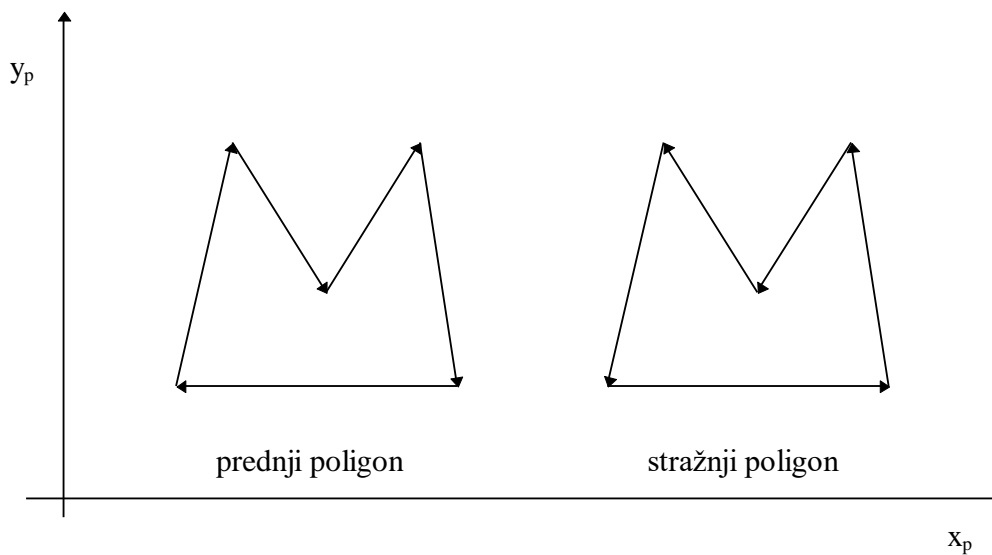
Slika 7.1. Stražnji poligoni (a) poligon je stražnji jer je kut veći od  $90^\circ$ ,  
(b) poligon nije stražnji jer je kut manji od  $90^\circ$ .

Kut  $\alpha$  između dva vektora  $\mathbf{a}$  i  $\mathbf{b}$  možemo odrediti na osnovi skalarnog produkta

$$\mathbf{ab} = |\mathbf{a}||\mathbf{b}|\cos\alpha \quad (7.1)$$

Drugi način je uklanjanje stražnjih poligona u sustavu prikaza, odnosno u ravnini projekcije. Pretpostavlja se da je redoslijed vrhova poligona u smjeru kazaljke na satu promatrano u sustavu scene i gledano izvan tijela. U ovom postupku, u ravnini projekcije redoslijed vrhova stražnjeg poligona je suprotan smjeru kazaljke na satu *slika 7.2*.

Neka je  $P$  popis vrhova poligona u sustavu prikaza,



Slika 7.2. Određivanje stražnjih poligona na osnovi orijentacije u projekciji.

$$P = (V_1 \ V_2 \ \dots \ V_n), \quad (7.2)$$

poligon  $P$  je stražnji poligon ako vrijedi

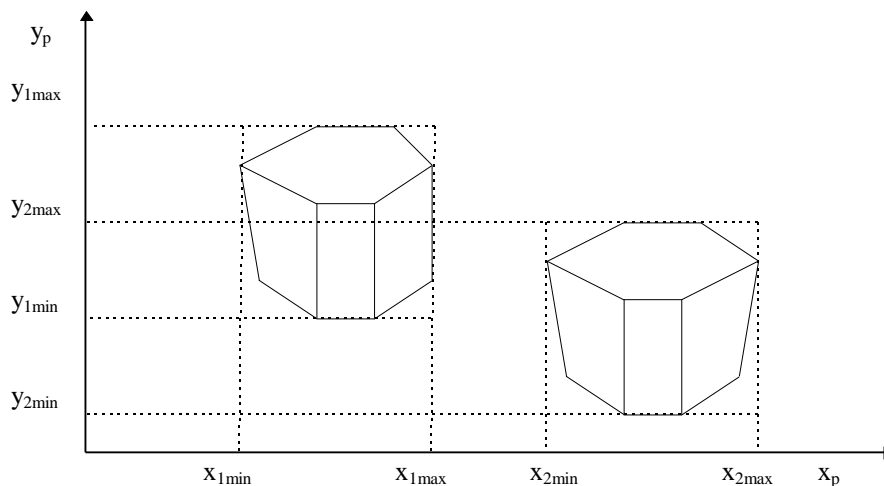
$$(\exists B_i) \left( (\forall_j) (B_i V_j) \geq 0, \quad i = 1 \dots n, \ j = 1 \dots n. \right) \quad (7.3)$$

### Minimaks provjere

Minimaks provjere izvode se u sustavu prikaza, odnosno ravnine projekcije. Moguće su provjere:

- tijelo-tijelo,
- poligon-tijelo,
- brid-tijelo,
- poligon-poligon,
- brid-poligon
- brid-brid.

Na slici 7.3. prikazana je minimaks provjera tijelo-tijelo.



Slika 7.3. Minimaks provjera za tijelo.

Za tijelo  $T_1$  i  $T_2$  treba odrediti minimalne i maksimalne  $x$ ,  $y$  koordinate. To su:

- $(x_{1min}, y_{1min})$   $(x_{1max}, y_{1max})$  za tijelo  $T_1$ ,
- $(x_{2min}, y_{2min})$   $(x_{2max}, y_{2max})$  za tijelo  $T_2$ .

Minimaks koordinate tijela određuju pravokutnike. Ako se pravokutnici u ravnini projekcije ne preklapaju tijelo  $T_1$  ne skriva tijelo  $T_2$ .

## 7.2. Raspon koordinata i položaj tijela

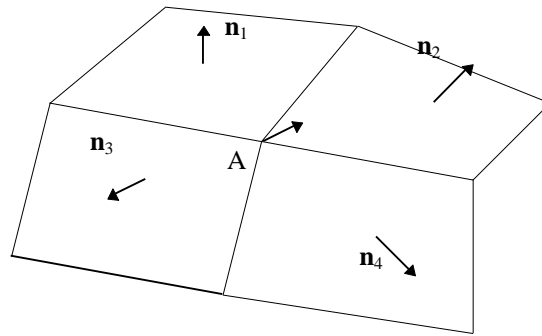
Tijelo može biti zadano u različitom rasponu koordinata i na različitom položaju u sceni. Minimaks koordinate tijela potrebne su pri normiranju veličine tijela na zadani radni prostor, te za pomak središta tijela u ishodište scene (ili neki određen položaj).

$$\begin{aligned} \text{velicina\_x} &= x_{\max} - x_{\min} & \text{srediste\_x} &= (x_{\max} + x_{\min})/2 \\ \text{velicina\_y} &= y_{\max} - y_{\min} & \text{srediste\_y} &= (y_{\max} + y_{\min})/2 \\ \text{velicina\_z} &= z_{\max} - z_{\min} & \text{srediste\_z} &= (z_{\max} + z_{\min})/2 \end{aligned}$$

Ako tijelo želimo smjestiti u radni prostor  $[-1, 1]$ , potrebno je pomaknuti tijelo za  $(-\text{srediste\_x}, -\text{srediste\_y}, -\text{srediste\_z})$ , te načiniti skaliranje s dvostrukom recipročnom vrijednošću najvećeg raspona, odnosno  $2/\max(\text{velicina\_x}, \text{velicina\_y}, \text{velicina\_z})$ .

Ukoliko unaprijed ne znamo položaj i raspon koordinata tijela, a želimo ga smjestiti u zadani radni prostor npr.  $[-1, 1]$  po prostornoj koordinati  $x$ ,  $y$  i  $z$ , postupiti ćemo prema opisanom postupku, kako ne bi morali mijenjati parametre transformacije pogleda i projekcije.

Ako scena sadrži više objekata, sličan potprogram može biti primijenjen na svako učitano tijelo (kako bi svi učitani objekti bili u istom rasponu), a zatim skaliramo i razmjestimo učitane objekte u danim okvirima scene.



Slika 7.4 Određivanje normale u vrhu A.

### 7.3. Određivanje vektora normale u vrhovima tijela

U modelu osvjetljavanja potrebno je poznavati normale u vrhovima tijela. Normale u vrhovima određuju se na osnovi normala poligona (Slika 7.4). Normale poligona moraju biti normirane (pojedine komponente podijeljene s korijenom zbroja kvadrata komponenti). U pojedinom vrhu načini se suma komponenti i podjeli brojem poligona koji su sudjelovali u određivanju normale.

$$\mathbf{n}_{Ax} = \frac{1}{4}(\mathbf{n}_{1x} + \mathbf{n}_{2x} + \mathbf{n}_{3x} + \mathbf{n}_{4x}), \quad (7.4)$$

gdje su  $\mathbf{n}_{1x}$ ,  $\mathbf{n}_{2x}$ ,  $\mathbf{n}_{3x}$ ,  $\mathbf{n}_{4x}$ , komponente  $x$  normala  $\mathbf{n}_1$ ,  $\mathbf{n}_2$ ,  $\mathbf{n}_3$ ,  $\mathbf{n}_4$ , koje sudjeluju u određivanju normale vrha. Na sličan način odrede se ostale komponente vektora normale u vrhu A.

### 7.4. Model osvjetljavanja

Na osnovi **modela osvjetljavanja** određujemo intenzitet u točki. Koristit ćemo Phongov model osvjetljavanja. Na model osvjetljavanja nadovezuje se postupak sjenčanja. Osnovni **postupci sjenčanja** su konstantno sjenčanje, Gouraudov postupak i Phongov postupak.

Phongov model osvjetljavanja čini jednostavnu aproksimaciju fizikalnog modela. Phongov model osvjetljenja sačinjava linearna kombinacija tri komponente: ambijentnu, difuznu i zrcalnu.

**Ambijentna komponenta.** Obično globalno izračunavanje doprinosa refleksije svih površina u sceni nije prisutno, pa ambijentna komponenta predstavlja aproksimaciju tih doprinosa. U slučaju kada ambijentna komponenta nije prisutna, poligoni koji su stražnji obzirom na izvor svjetlosti će biti crni. Kako uvijek postoji refleksivnost površina okoline navedene plohe nisu sasvim crne. Kako ti poligoni ne moraju biti stražnji za promatrača bit će vidljivi. Ambijentna komponenta određena je

$$I_g = I_a k_a \quad (7.5)$$

gdje je  $I_a$  intenzitet koji želimo na zaklonjenim plohama (0-255), a  $k_a$  koeficijent ( $0 \leq k_a \leq 1$ ).

**Difuzna komponenta** je

$$I_d = I_i k_d \cos \varphi = I_i k_d (\mathbf{L}\mathbf{N}), \quad 0 \leq \varphi \leq \pi, \quad (7.6)$$

gdje je  $I_i$  intenzitet točkastog izvora,  $\varphi$  kut između normale na površinu i vektora prema izvoru,  $k_d$  je empirijski koeficijent refleksije ( $0 \leq k_d \leq 1$ ).  $\mathbf{L}\mathbf{N}$  označava skalarni produkt normiranih vektora  $|\mathbf{L}| = 1$  i  $|\mathbf{N}| = 1$ . Kada je kut  $\varphi$  veći od  $90^\circ$  skalarni produkt je negativan, što znači da je

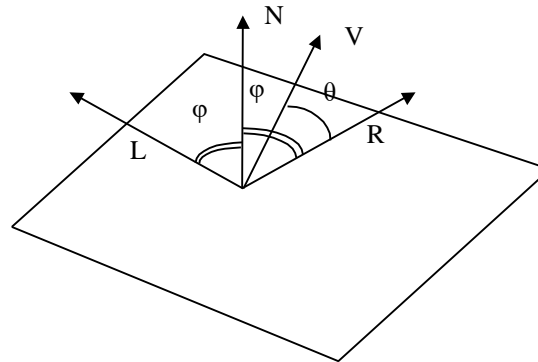
ploha stražnja (neosvijetljena) za zadani izvor. U tom slučaju se uzima vrijednost nula za difuznu komponentu. Ukupan doprinos ambijentne i difuzne komponente je:

$$I_d = I_a k_a + I_i k_d (\mathbf{LN}), \quad 0 \leq \varphi \leq \pi, \quad (7.7)$$

Izvor svjetla



očište



Slika 7.4. Položaj normale  $\mathbf{N}$ , vektora prema izvoru  $\mathbf{L}$ , vektora prema promatraču  $\mathbf{V}$  i reflektirane zrake  $\mathbf{R}$ .

**Zrcalna komponenta** ovisi o kutu  $\theta$  između reflektirane zrake  $\mathbf{R}$  i zrake prema promatraču  $\mathbf{V}$

$$I_s = I_i k_s \cos^n \theta = I_i k_s (\mathbf{RV})^n \quad (7.8)$$

gdje je  $n$  indeks hrapavosti površine  $n \in (0, \infty)$ . Reflektirana zrake leži u ravnini određenoj normalom i upadnom zrakom, te zatvara kut  $\varphi$  sa normalom slika 7.4.

Ukupan doprinos određen je formulom

$$I = I_a k_a + I_i (k_d (\mathbf{LN}) + k_s (\mathbf{RV})^n) \quad (7.9)$$

U navedeni model možemo uvesti i ovisnost o udaljenosti objekta od izvora. Zbog pojednostavljenja za izvor se uzima da je u beskonačnosti ( $\mathbf{L}$  je konstanta), pa se za udaljenost obično uzima udaljenost  $r$  od promatrača. Uvođenje udaljenosti rezultira formulom

$$I = I_a k_a + \frac{I_i}{r + k} (k_d (\mathbf{LN}) + k_s (\mathbf{RV})^n) \quad (7.10)$$

Prema danoj formuli intenzitet se računa posebno za  $R$ ,  $G$ ,  $B$  komponente  $I_r$ ,  $I_g$ ,  $I_b$ , no u okviru vježbi dovoljno je implementirati ambijentnu i difuznu komponentu, bez

ovisnosti o udaljenosti, te računati intenzitet samo jedne komponente (npr.  $I_r$ ).

Parametri u formuli za računanje intenziteta odabiru se tako da ukupni intenzitet bude u zadanom rasponu. Na primjer, u rasponu od 0 do 1 ili 0 do 255. Koeficijenti  $k_s$  i  $k_d$  se zadaju od 0 do 1, skalarni produkt vektora prema izvoru i normale  $\mathbf{LN}$  u slučaju kada su vektori normirani je u rasponu 0 do 1. Ako postoji više izvora svjetlosti, zadavanje raspona može biti problem, odnosno rezultat može premašiti zadane okvire. Ako rezultat pređe postavljeni maksimum tada se pridružuje vrijednost maksimuma. Drugi način je izračunavanje svih vrijednosti intenziteta slike. Kada znamo sve vrijednosti intenziteta, poznat nam je i maksimum, pa prema tome možemo pridružiti linearno (ili nelinearno) intenzitete prema postavljenom rasponu. (Ovaj postupak nije pogodan u slučaju animacije, odnosno promjene položaja izora.)

### 7.5. Postupci sjenčanja

Osnovni postupci sjenčanja su konstantno sjenčanje, Gouraudov postupak i Phongov postupak.

**Konstantno sjenčanje** najjednostavnije je i koristi normalu poligona za izračunavanje intenziteta osvjetljenja poligona. Cijeli poligon ima isti intenzitet određen na osnovi modela osvjetljenja (formula 7.8. ili 7.9). Za trokut to znači da će tri vrha imati pridruženu istu boju.

**Gouraudov postupak sjenčanja** koristi izračunavanje intenziteta u vrhovima na osnovi normala u vrhovima. Tako dobiveni intenzitet linearno se interpolira na poligonu. To znači da ćemo na osnovi tri različite normale dobivene po formuli (7.4) u vrhovima trokuta odrediti intenzitete u vrhovima.

**Phongov postupak sjenčanja** prvo interpolira izračunate normale vrhova poligona, kako bi se za svaki slikovni element u prikazu odredila normala, a zatim, za svaki slikovni element poligona u ravnini projekcije se računa intenzitet prema modelu osvjetljenja.

## 7.6. Radni zadatak

Uz radni zadatak priložen je primjer koji crta trokut zadan 3D koordinatama i intenzitetima boje u vrhovima. Umjesto trokuta potrebno je prikazati učitano tijelo uz različite načine sjenčanja i uklanjanje stražnjih poligona.

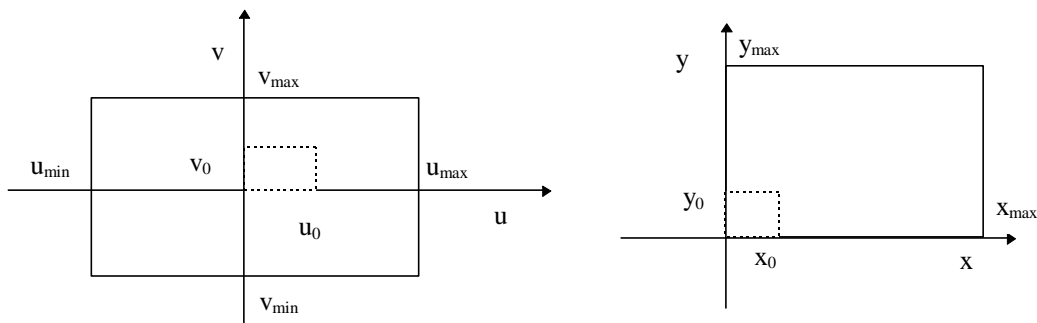
1. Učitati tijelo prema postupku opisanom u vježbi 4. Prema točki 7.2. smjestiti tijelo u radni prostor  $[-1, 1]$ . Načiniti transformaciju pogleda i perspektivnu projekciju prema vježbi 5, te usporediti rezultat ako se koristi ponuđeni kod u OpenGL-u.
2. Odrediti stražnje poligone na tijelu prema slici 7.1 i formuli 7.1.
3. Prikazati tijelo u žičnoj formi uz uklanjanje stražnjih poligona.
4. Zadati položaj izvora.
5. Na osnovi normala poligona i vektor prema izvoru  $\mathbf{L}$  iz središta poligona, prema formuli 7.7, odrediti intenzitete poligona (ambijentnu i difuznu komponentu).
6. Prikazati tijelo tako da sva tri vrha poligona imaju isti intenzitet određen intenzitetom poligona (konstantno sjenčanje).
7. Odrediti normale u vrhovima za pojedine površine prema formuli 7.4. Normirati normale.
8. Na osnovi normala u vrhovima i vektora iz vrhova prema izvoru, prema formuli 7.7, odrediti intenzitete u vrhovima (ambijentnu i difuznu komponentu).
9. Prikazati tijelo tako da sva tri vrha imaju različite intenzitete određene u koraku 8. (Gouraud-ovo sjenčanje).



## 8. Fraktali – Mandelbrotov i Julijev fraktalni skup

### 8.1. Kompleksna ravnina i ravnina prikaza

Funkcija kompleksne varijable  $f(z_n)$  promatra se u kompleksnoj ravnini čije su osi  $(u, v)$ . Ravnina prikaza  $(x, y)$  je ravnina u kojoj prikazujemo promatranu kompleksnu funkciju. Prevođenje iz sustava  $(O u v)$  u sustav  $(O' x y)$  ovisi o promatranom području u pojedinim sustavima. Neka je promatrano područje kompleksne funkcije zadano s  $u_{\max}$ ,  $u_{\min}$ ,  $v_{\max}$  i  $v_{\min}$ . Područje sustava prikaza neka je zadano s  $x_{\max}$  i  $x_{\min}$  (Slika 8.1).



Slika 8.1. Ravnina kompleksne funkcije i ravnina prikaza.

Sustav prikaza je zaslon, pa su vrijednosti na  $x$  i  $y$  osi diskretne. Koordinate točke  $u_0$  i  $v_0$  u kompleksnoj ravnini koje odgovaraju vrijednostima  $x_0$  i  $y_0$  su:

$$u_0 = \frac{u_{\max} - u_{\min}}{x_{\max}} x_0 + u_{\min}, \quad v_0 = \frac{v_{\max} - v_{\min}}{y_{\max}} y_0 + v_{\min} \quad (1)$$

Navedenim izrazima definirano je prevođenje iz jednog u drugi sustav.

### 8.2. Skupovi Mandelbrota i Julije

Neka je zadano iterativno preslikavanje:

$$z_{n+1} = f(z_n), \quad (2)$$

gdje je  $f(z_n)$  na primjer  $z_{n+1} = f(z_n) = z_n^2 + c$ ,  $z, c \in \mathbb{C}$ , a  $c$  je odabrana točka kompleksne ravnine za koju ispitujemo konvergenciju generiranog niza. Za ovako definirano iterativno preslikavanje možemo promatrati da li niz koji generiramo  $(z_0, z_1, z_2, \dots)$  konvergira ili ne. Uvjet zaustavljanja u programskoj implementaciji može biti različit. Jedan primjer kriterija kojim ustanovljavamo da li niz konvergira je ocjena apsolutne vrijednosti:

$$|z_n| = \sqrt{u^2 + v^2}, |z_n| < \varepsilon, \quad n > n_0$$

Ako iterativno preslikavanje  $z_{n+1} = f(z_n)$  nakon  $n$  iteracija ne zadovolji uvjet  $|z| > \varepsilon$  reći ćemo da niz konvergira, a inače da divergira. Definirat ćemo “brzinu divergencije” brojem iteracija koje su potrebne da uvjet  $|z| > \varepsilon$  bude zadovoljen. Postupak se provodi tako da se za svaki slikovni element ravnine prikaza  $(x_0, y_0)$  odredi pripadna točka kompleksne ravnine, te za nju

ispita konvergencija pripadnog niza. Područje kompleksne ravnine unutar kojega iterativno preslikavanje generira konvergentne nizove naziva se Mandelbrot-ov skup.

Za Julijev skup potrebno je odabrati  $c \in \mathbb{C}$  (točku kompleksne ravnine), a  $z_0$  je točka kompleksne ravnine za koju ispitujemo konvergenciju niza. Ako se za  $c \in \mathbb{C}$  odabere točka unutar Mandelbrot-ovog skupa Julijev skup će biti povezan, a inače nepovezan.

### 8.3. Radni zadatak

Postupak za Mandelbrotov skup:

1. Učitati prag epsilon  $eps$  i maksimalan broj iteracija  $m$ .
2. Učitati područje kompleksne ravnine koja se promatra  $(u_{\min}, u_{\max}), (v_{\min}, v_{\max})$ .
3. Pročitati razlučivost zaslona  $x_{\max}, y_{\max}$ .
4. Za svaku točku zaslona  $x_0, y_0$ :
  - a) odrediti  $u_0, v_0$  (prema formuli 1).
  - a) Postaviti:  $k = -1, c_{\text{real}} = u_0, c_{\text{imag}} = v_0, z_0 = 0$ .
  - b) Činiti:

$$k = k + 1,$$

$$z_{n+1} = z_n^2 + c$$

$$r = \sqrt{z_{\text{real}}^2 + z_{\text{imag}}^2}$$

dok je ispunjen uvjet  $r < eps$  i  $k < m$ :

5. Na mjestu  $x_0, y_0$  iscrtati slikovni element u boji  $k$ .

Primjer:  $eps=100, m=16, (u_{\min}, u_{\max}) = (-1.5, 0.5), (v_{\min}, v_{\max}) = (-1, 1)$

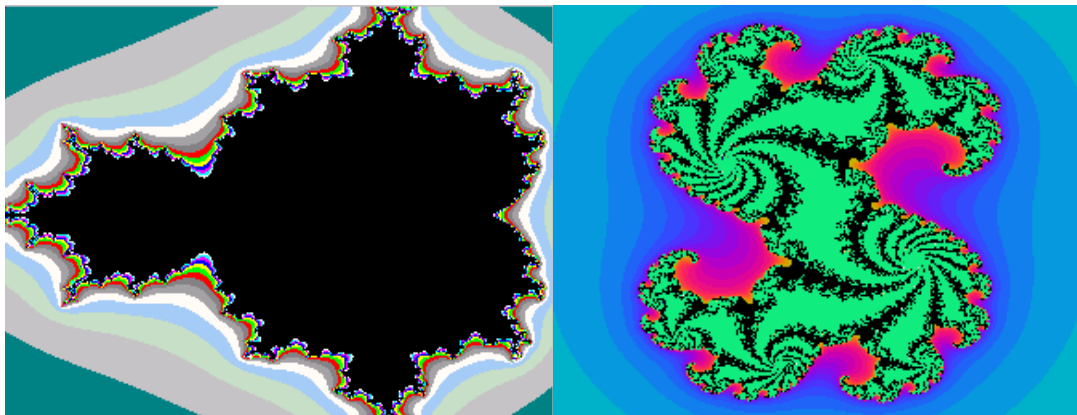
Postupak za Julijev skup:

Postupak je sličan prethodnom, a promjene su:

1. Dodatno učitati i kompleksnu konstantu  $c \in \mathbb{C}$ .

- a) Postaviti:  $k = -1, z_{\text{real}} = u_0, z_{\text{imag}} = v_0$ .

Primjer:  $eps=100, m=16, (u_{\min}, u_{\max}) = (-1, 1), (v_{\min}, v_{\max}) = (-1.2, 1.2), (c_{\text{real}}, c_{\text{imag}}) = (0.32, 0.043)$ .



Slika 8.2: Mandelbrotov i Julijev fraktalni skup.