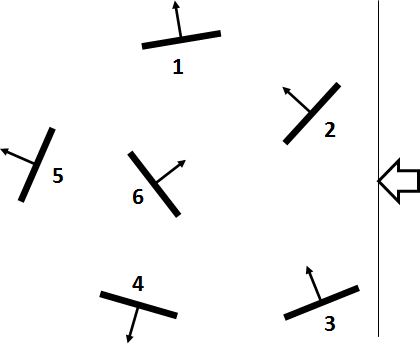
**Računarska grafika – Usmeni** Ime i prezime:

02.03.2021. Broj indeksa:

1. Algoritam za uklanjanje skrivenih površina zasnovan je na BSP stablu. Ukoliko je potrebno iscrtati scenu koja se sastoji od šest poligona, prikazanih na slici, nacrtati kako izgleda BSP stablo i napisati kojim redosledom se icrtavaju poligoni, ako se posmatrač nalazi sa desne strane (pozicija strelice), a poligoni za podelu prostora se biraju redosledom tako da je sledeći sa:
2. najvećim indeksom (kreće se od 6),
3. najmanjim indeksom (kreće se od 1).

a)

Obilazak:

b)

Obilazak:

1. Navesti **modele senki**, i za svaki od njih navesti glavnu prednost.
   1. Naziv: Prednost:
   2. Naziv: Prednost:
   3. Naziv: Prednost:
   4. Naziv: Prednost:

Poređati modele senki po složenosti iscrtavanja (od najjednostavnijeg ka složenijim): Poređati modele senki po realističnosti (od najrealističnijeg ka manje realističnim):

1. Neka su zadati poligon i duž svojim temenima {(1,2), (3,0), (5,2), (3,4)} i {(0,2), (5,4)}, respektivno. Korišćenjem **Cyrus-Back** algoritma, odrediti sve potencijalne tačke preseka, tačnije njihove vrednosti parametra t. Za prave tačke preseka odrediti (x,y) koordinate i navesti razlog zašto su od svih potencijalnih tačaka preseka izabrane baš te.

t0 = , t1 = , t2 = , t3 = , t4 = . Tačke preseka su: P0 = ( , ) P1 = ( , ) zato što

(Formula za računanje preseka: t = NL  [P0 – PL] / -NL  [P1 - P0] )

1. Na posebnom listu napisati implementaciju funkcije *Ellipse(CDC\* pDC, int x1, int y1, int x2, int y2, COLORREF col)*, koja na što efikasniji i precizniji način iscrtava elipsu, debljine 1 piksel, u zadatom DC-u. Za postavljanje odgovarajućeg piksela na boju *col*, koristiti GDI funkciju *SetPixel(int x, int y, COLORREF col)*.
2. Izvesti (na posebnom papiru) matricu za **kabinet** projekciju, ako ugao koji u projekciji zaklapaju stranice upravne na projekcionu ravan sa X-osom iznosi **30**. Vrednosti upisati ispod zadatka, na tekućoj stranici. Smatrati da projekciona matrica množi koordinate temena sa leve strane (vp = P  v).
3. Ako su zadate sledeće dve OpenGL komande: **glViewport(x0, y0, w, h)** i **gluOrtho2D(left, right, bottom, top)**, napisati kako izgleda transformaciona matrica koja vrši preslikavanje prozora u zaslon (*window to viewport*). Smatrati da transformaciona matrica množi koordinate temena sa leve strane (vp = Mv). Prvo napisati vrednost matrice kao kompoziciju elementarnih transformacija, a zatim množenjem matrica dobiti jedinstvenu transformacionu matricu.

Kompozicija elementarnih transformacija:

Konačna matrica:

7. Poligon je zadat sledećom listom temena: { (0,2), (2,0), (4,0), (4,2), (6,2), (8,4), (6,6), (4,6), (4,4) , (0,4) }. Nacrtati kako izgleda **tabela ivica** (ET) i **tabela aktivnih ivica** (AET) kod odgovarajućeg algoritma za ispunu poligona, za sve celobrojne vrednosti Y iz opsega [0,6].

Predmetni nastavnik