Graficki API-i:

1. Definisati graficki API. Nabrojati i objasniti osnovne grupe funkcija koje bi svaki dobar graficki API trebalo da ima.

2(x2). Windows GDI - opis i princip rada. Objasniti device context.

3. Šta je grafički API?

**Hardver racunarske grafike:**

1. Navesti sve logicke tipove ulaznih uredjaja i sve tipove izlaznih uredjaja. Objasniti namenu svakog tipa uredjaja.

2(x3). Objasniti iscrtvanje ekrana sa preplitanjem i bez preplitanja.

3. Objasniti razliku izmedju rasterskih i vektorskih podataka. Opisati nacine rada rasterskih i vektorskih grafickih uredjaja.

4. Uporediti LCD ili DLP video projektore, upisivanjem odgovarajućeg tipa u tablicu ispod.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| kvalitetnije boje | manje dimenzije | jeftiniji | veća buka | veći kontrast |
| LCD | DLP | LCD | DLP | DLP |

**Rasterizacija:**

1(x4). Izvesti i dati realizaciju Bresenham-ovog algortima za crtanje linije.

2. Izvesti i dati realizaciju Bresenham-ovog algortima za crtanje kruznice.

3(x3). Izvesti i dati realizaciju diferencijalnog algoritma I reda za skaniranje elipse.

4. Izvesti i dati realizaciju diferencijalnog algoritma II reda za skaniranje elipse.

5. Izvesti i dati realizaciju Bresenham-ovog algortima za crtanje linije. Naci piksele koji ce biti selektovani prilikom crtanja linije (4, 4), (8, 16).

6. Napisati funkciju kojom se iscrtava pravolinijski segment između tačaka (x0, y0) i (x1, y1), korišćenjem nagibnog algoritma. Funkcija mora da iscrtava kontinuirani niz piksela, bez obzira na nagib prave.

7. 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)**.

8. Napisati funkciju **Ellipse(int x1, int y1, int x2, int y2)**, za crtanje elipse korišćenjem trigonometrijske metode. Parametri se zadaju na isti način kao kod odgovarajuće GDI funkcije.

**Popunjavanje objekata:**

1. Napisati algoritam za popunu poligona zadatog listom temena.

2(x2). Poligon je zadat sledećom listom temena: { (1,0), (3,0), (4,1), (3,3), (2,3), (1,5), (0,3) }. Nacrtati kako izgleda tabela ivica (ET) i tabela aktivnih ivica (AET) kod odgovarajućeg algoritma za ispunu poligona.

3. 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].

**Odsecanje objekata:**

1. Objasniti Cohen-Sutherland-ov algoritam za odsecanje linija i dati realizaciju u C/C++.

2(x2). Primenom Cohen-Sutherland-ovog algoritma odsecanja linija odrediti presečne tačke i odgovarajuće kodove, pri traženju preseka linije zadate krajnjim tačkama (0, 250) i (300,50), i prozora sa koordinatama (100,100) i (200,200). Smatrati da je redosled ispitivanja ivica: TOP, BOTTOM, RIGHT, LEFT.

a. Zašto se linija trivijalno ne prihvata? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. Zašto se linija trivijalno ne odbacuje? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

c. Navesti koordinate svih karakterističnih tačaka (počevši od krajnjih tačaka linije) i njihove odgovarajuće kodove. L predstavlja koordinate leve tačke tekućeg linijskog segmenta, a R desnog. U uglastim zagradama upisati kod. Popuna ide sleva udesno.

|  |  |  |
| --- | --- | --- |
| L ( , ) → [ ] | L ( , ) → [ ] | L ( , ) → [ ] |
| R ( , ) → [ ] | R ( , ) → [ ] | R ( , ) → [ ] |
| L ( , ) → [ ] | L ( , ) → [ ] | L ( , ) → [ ] |
| R ( , ) → [ ] | R ( , ) → [ ] | R ( , ) → [ ] |

3. 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])

**Geometrijske transformacije:**

1. Izracunati matricu za 2D rotaciju za ugao od 45 stepeni oko tacke P(5, 3).

2. Izracunati matricu za 2D rotaciju za ugao od 135 stepeni oko tacke P(3, 8).

3. Izracunati matricu za 2D refleksiju u odnosu na pravu x + y = 0.

4(x2). Izracunati matricu za 2D refleksiju u odnosu na pravu y - x = 0.

5. Izracunati matricu za 2D refleksiju u odnosu na pravu 2\*y - 2\*x = 0.

6. Izracunati matricu za 2D refleksiju u odnosu na pravu y = -x.

7. Izvesti transformacionu matricu za rotaciju oko tacke (Dx, Dy) za ugao od -90 stepeni.

8. Izvesti (na posebnom papiru) matricu za kompozitnu 2D transformaciju koja se sastoji od rotacije oko tačke (2,2) za 90°, zatim translacije za (4,0) i ponovo rotacije za 90° oko tačke (8, 4). Smatrati da matrice množe vektor koordinata sa leve strane. Napisati prvo kompoziciju elementarnih transformacija, a zatim svesti rezultat na jednu matricu.

9. Izvesti matricu za refleksiju u odnosu na osu y = x + 2. Matricu izvesti na posebnom listu, a u nastavku napisati samo konačni oblik transformacione matrice. Smatrati da matrica množi vektor koordinata sa leve strane ( v’ = M ⋅ v).

**Geometrijske projekcije:**

1. Izvesti matricu za perspektivnu projekciju ukoliko se centar projekcije nalazi u koordinatnom pocetku a projekcioana ravan je ravan z = 2.

2. Izvesti (na posebnom papiru) matricu za perspektivnu projekciju (vrednosti upisati ispod zadatka, na tekućoj stranici), kod koje je posmatrač u koordinatnom početku i gleda u pravcu negativne Z ose, a projekciona ravan se nalazi na rastojanju d od posmatrača. Smatrati da projekciona matrica množi koordinate temena sa leve strane (vp = P ⋅ v).

3. 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).

4. Izvesti transformacionu matricu za „kavaljersku“ projekciju. Smatrati da u projekciji stranice upravne na projekcionu ravan zaklapaju ugao od 30° sa X-osom. Matricu izvesti na posebnom listu, a u nastavku napisati samo konačni oblik transformacione matrice. Smatrati da matrica množi vektor koordinata sa leve strane ( v’ = M ⋅ v).

**2D i 3D pogled:**

1. 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 = Mxv). Prvo napisati vrednost matrice kao kompoziciju elementarnih transformacija, a zatim množenjem matrica dobiti jedinstvenu transformacionu matricu.

2. 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.

**Algoritmi za ostvarivanje realnosti prikaza:**

1. Navesti algoritme za uklanjanje sakrivenih ivica i povrsi. Objasniti Z-bafer algoritam. Navesti dobre i lose strane ovog algoritma.

2. Za 3D scenu poligona na slici formirati BSP stalbo u skladu sa algoritmom za uklanjanje sakrivenih ivica i povrsi koji korisit BSP stabla. Navesti redosled iscrtvanja poligona za smer posmatranja dat na slici.

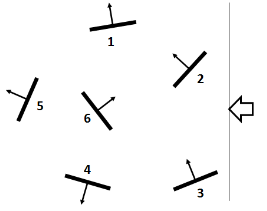
3. Navesti algoritme za uklanjanje sakrivenih ivica i povrsi. Objasniti Warnock-ov i Watkinson-ov algoritam.

4. Navesti 5 testova preklapanja kod slikarevog algoritma.

5. 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:

a) najvećim indeksom (kreće se od 6)

b) najmanjim indeksom (kreće se od 1).



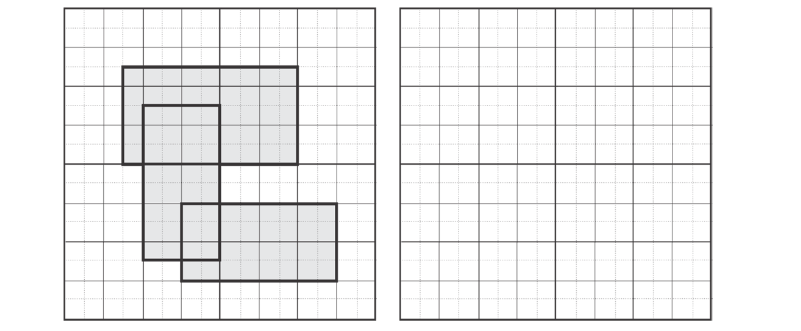
b)

a)

Obilazak:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Obilazak:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6. Ako se uklanjanje skrivenih površina koristi Warnock-ov algoritam, označiti na slici redni broj iteracije u kojoj će biti iscrtan odgovarajući kvadrant. Smatrati da nulta iteracija treba da iscrta čitav prozor.



**RG-12 (Boje u racunarskoj grafici):**

1. Navesti modele boja koji se koriste u racunarskoj grafici. Koje vrednosti po komponentama ima boja u CMY modelu, ukoliko u RGB modelu ima vrednosti (233, 148, 228)? Vrednosti po komponentama su iz opsega 0..255.

2. Navesti modele boja koji se koriste u racunarskoj grafici. Koje vrednosti po komponentama ima boja u CMY modelu, ukoliko u RGB modelu ima vrednosti (230, 145, 248)? Vrednosti po komponentama su iz opsega 0..255.

3. Ako je zadata CMY boja (0.5, 1.0, 0.0), napisati koje vrednosti (po kanalima, max. vrednost 255) ona ima u RGB modelu boja, kao i koja je to boja:

4. Ako je zadata narandžasta boja, sa RGB vrednostima (245,128,64), napisati koje vrednosti ona ima u CMY i CMYK modelu boja.

**RG-13 (Senke, senčenje i bojenje):**

1. Navesti modele senki koji se koriste u racunarskoj grafici. Detaljno objasniti model koji koristi lazne senke. Sta su dobre a sta lose strane ovog modela?

2(x2). Navesti modele senki koji se koriste u racunarskoj grafici. Detaljno objasniti Flat model. Sta su dobre a sta lose strane ovog modela?

3(x2). Navesti modele senki koji se koriste u racunarskoj grafici. Detaljno objasniti Phong model. Sta su dobre a sta lose strane ovog modela?

4. Navesti modele senki koji se koriste u racunarskoj grafici. Detaljno objasniti Gouraud-ov model. Sta su dobre a sta lose strane ovog modela?

5. Koje grupe algoritama za sencenje postoje? Koji algoritmi pripadaju kojoj grupi? Objasniti Flat algoritam. (24, 25, 26 pitanje u skripti)

6. Navesti modele senki, i za svaki od njih navesti glavnu prednost.

a. Naziv:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Prednost:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. Naziv:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Prednost:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

c. Naziv:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Prednost:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

d. Naziv:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Prednost:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Poređati modele senki po složenosti iscrtavanja (od najjednostavnijeg ka složenijim):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Poređati modele senki po realističnosti (od najrealističnijeg ka manje realističnim):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**RG-14 (Modeliranje geometrijskih formi):**

1. Pravu koja je odredjena tackama A(2, 2) i B(5, 8) predstaviti u eksplicitnom, implicitnom i parametarskom obliku. (x2)

2. Pravu koja je odredjena tackama A(4, 4) i B(10, 16) predstaviti u eksplicitnom, implicitnom i parametarskom obliku.

3. Pravu koja je odredjena tackama A(4, 4) i B(7, 10) predstaviti u eksplicitnom, implicitnom i parametarskom obliku.