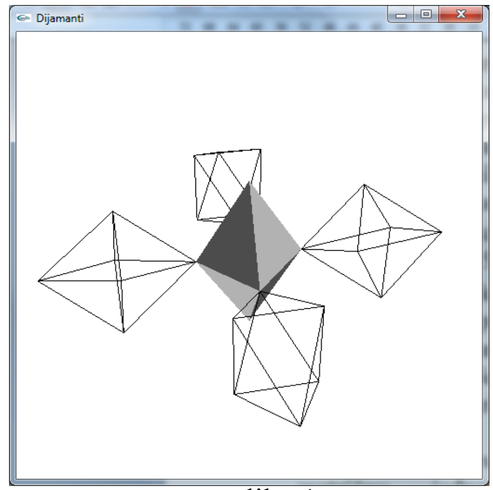
ELEKTROTEHNIČKI FAKULTET 04.05.2011.
Katedra za RTI

RAČUNARSKA GRAFIKA

drugi kolokvijum

1) [25] **OpenGL:** Napisati na jeziku C ili C++ deo programa za crtanje scene prikazane na slici 1 primenom grafičke biblioteke OpenGL. Scena se sastoji od 5 identičnih geometrijskih tela. Jedno geometrijsko telo čine dve podudarne piramide kvadratne osnove, koje su oslonjene osnovama jedna na drugu. Telo u centru scene se crta popunjeno, svetlom i tamnom sivom (videti sliku). Ostala 4 tela su prozirna i nalaze se u odnosu na centralno telo kao na slici 1. Tela rotiraju oko vertikalne ose koja je podudarna sa vertikalnom osom simetrije centralnog tela, a tekuća vrednost ugla rotacije se nalazi u globalnoj promenljivoj deklarisanoj na sledeći način: `double ugao;`



slika 1

Koristi se projekcija sa perspektivom sledećih parametara: ugao vidnog polja = 30°; odnos širine i visine = 1; bliža ravan odsecanja = 1; dalja ravan odsecanja = 30. Napisati posebnu funkciju koja vrši osnovnu inicijalizaciju OpenGL sistema potrebnu za crtanje scene. Posmatračku kameru približno postaviti u položaj koji bi proizveo prikazanu sliku. Smatrati da je otvaranje prozora za crtanje realizovano u glavnom programu koji nije potrebno pisati. Takođe treba smatrati da se u pravilnim vremenskim intervalima vrši osvežavanje scene, kada se ažurira vrednost promenljive ugao, što takođe nije potrebno pisati. **Napomena**: rezultujuća slika ne sme da zavisi od redosleda crtanja elemenata scene.

2) [25] Postaviti jednu matričnu jednačinu koja određuje projekciju sa perspektivom iz centra projekcije P(2,10,5) na projekcionu ravan $x+2z=2$ slike proizvoljne tačke (u desnom 3D koordinatnom sistemu) u ogledalu postavljenom na ravan $x+y=-4$. U sve matice elementarnih transformacija koje učestvuju u matričnoj jednačini uvrstiti konkretne vrednosti. Nije potrebno množiti matrice.

3) [50] Odgovoriti koncizno (jedna do dve rečenice) na sledeća pitanja:

- **a)** Definisati pojam iščezavajuće tačke.
- **b)** Koju transformaciju (između kojih koordinatnih sistema) određuje matrica orijentacije prikaza (*view orientation*) u SPHIGS paketu?
- **c)** Da li se atributi SPHIGS strukture primenjuju na podstrukturu? Za koje atribute je to naročito važno i zašto?
- **d)** Koji načini brisanja elemenata iz SPHIGS strukture postoje i koji se elementi brišu odgovarajućim načinom?
- **e)** Kako se u SPHIGS paketu obezbeđuje jednoznačnost identifikacije pri selekciji (pik-korelaciji) struktura?

Napomene:

1. Kolokvijum traje 120 minuta.
2. Nije dozvoljena upotreba literature niti programabilnih kalkulatora.
3. Dozvoljena je upotreba OpenGL podsetnika.

Rešenja zadataka
drugi kolokvijum 2011

1) Rešenje

```
void crtajPiramidu(float *boja1, float *boja2)
{
  glPushMatrix();
  for(int i = 0; i < 4; i++)
  {
    if( i & 1 )
      glColor3fv(boja1);
    else
      glColor3fv(boja2);

    glBegin(GL_TRIANGLES);
            glVertex3f(0,1,0);
            glVertex3f(-0.5f, 0, 0.5f);
            glVertex3f(0.5f, 0, 0.5f);
    glEnd();

    glRotated(90, 0, 1, 0);
  }
  glPopMatrix();
}

void crtajDijamant(float *boja1, float *boja2)
{
  glPushMatrix();
    crtajPiramidu(boja1, boja2);
    glScalef(1,-1,1);
    glFrontFace(GL_CW);
    crtajPiramidu(boja2, boja1);
    glFrontFace(GL_CCW);
  glPopMatrix();
}

void crtajFiguru()
{
    static float crna[] = { 0.f, 0.f, 0.f };
    static float tamno_siva[] = { 0.3f, 0.3f, 0.3f };
    static float svetlo_siva[] = { 0.7f, 0.7f, 0.7f };

    glPushMatrix();
    glPushAttrib(GL_ALL_ATTRIB_BITS);

    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    crtajDijamant(tamno_siva, svetlo_siva);

    glDisable(GL_CULL_FACE);
    glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);
    glRotatef(90, 1, 0, 0);
    for(int i = 0; i < 4; i++)
    {
      glPushMatrix();
        glRotatef(i*90+45, 0, 0, 1);
        glTranslatef(0, 1+SQRT2/2, 0);
        crtajDijamant(crna, crna);
      glPopMatrix();
    }
    glPopAttrib();
    glPopMatrix();
}


void crtajScenu(void)
{
  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
  glLoadIdentity();
  gluLookAt(-6, 5, 4, 0, 0, 0, 0, 1 ,0);

  glRotated(angle, 0, 1, 0);
  crtajFiguru();

  glFlush();
}


void init()
{
  glClearColor (1.0, 1.0, 1.0, 0.0);
  glEnable(GL_CULL_FACE);
  glFrontFace(GL_CCW);

  glEnable(GL_DEPTH_TEST);
  glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);

  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();
  gluPerspective(30, 1, 1, 30);

  glMatrixMode(GL_MODELVIEW);
  glLoadIdentity();
}
```

2) Rešenje

[X' Y' Z' W] = [X Y Z 1] * M

[X" Y" Z" 1] = $[\frac{X'}{W} \ \frac{Y'}{W} \ \frac{Z'}{W} \ 1]$

M = Translacija1 * Rotacija1 * Ogledanje * Rotacija2 * Translacija2 * Rotacija3 * Projekcija * Rotacija4 * Translacija3

Translacija1 = T(-4,0,0) = $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 1 \end{bmatrix}$

Rotacija1 = Rz($\frac{\pi}{4}$) = $\begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & 0 \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Ogledanje = Oyz = $\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Rotacija2 = Rz($-\frac{\pi}{4}$) = $\begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & 0 \\ -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Translacija2 = T(4,10,1)= $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -4 & -10 & -1 & 1 \end{bmatrix}$

$\alpha = arctg(\frac{1}{2})$

Rotacija3 = Ry(α) = $\begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Projekcija = Pp(d= $\sqrt{20}$) = $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\sqrt{20}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Rotacija4 = Ry(- α) = $\begin{bmatrix} \cos(\alpha) & 0 & -\sin(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Translacija3 = T(0,-10,-1)= $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 10 & 1 & 1 \end{bmatrix}$