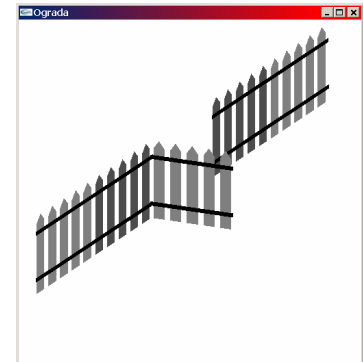


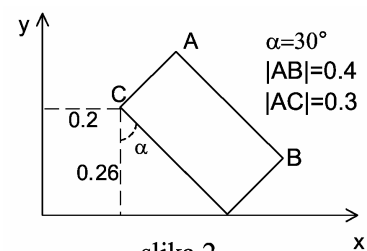
RAČUNARSKA GRAFIKA drugi kolokvijum

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 segmenata ograde. Svaki segment se sastoji od 5 stubova. Svi stubovi unutar jednog segmenta obojeni su na isti način, svetlo sivom ili tamno sivom. Stubove i segmente međusobno povezuju crne šipke koje se predstavljaju linijama debljine 5. Srednji segment ograde predstavlja vrata, otvorena pod uglom 60° . Koristi se ortografska projekcija. 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. **Napomena:** rezultujuća slika ne sme da zavisi od redosleda crtanja elemenata scene.



slika 1

2) [25] Posmatraju se objekti u desnom 3D koordinatnom sistemu. Nakon ogledanja u ogledalu postavljenom na ravan $x=y+5$, vrši se projekcija sa perspektivom za posmatrača koji se nalazi u tački $P(-2, 0, 0)$ na ravan $z=y-4$. Delovi projekcije objekata koji se nalaze u prozoru definisanom sa $x \in [-4, 4]$, $y \in [-3.5, 3.5]$ se potom preslikavaju u prikazni prozor definisan slikom 2, tako da se stranica prozora $y=-3.5$ preslika u stranicu AB, pri čemu se tačka prozora $(-4, -3.5)$ preslikava u tačku B. Postaviti matricnu jednačinu preslikavanja iz 3D prostora u zadati prozor. U sve matrice elementarnih transformacija uvrstiti konkretne vrednosti. Nije potrebno množiti matrice.



slika 2

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

- Navesti elementarnu matricu rotacije desnog 3D pravouglonog koordinatnog sistema oko Y-ose.
- Da li se mogu jednostavno meriti dužine u ravnima paralelnim projekcionoj ravni kod projekcije sa perspektivom i zašto?
- Da li se u SPHIGS-u za formiranje kompozitne matrice sa ciljem da se podrži zadati redosled transformacija koristi pre- ili post-konkatenacija i zašto?
- Ako se promeni atribut SPHIGS podstrukture, da li se po povratku iz podstrukture primenjuje promenjena vrednost u podstrukтури, ili stara vrednost atributa strukture?
- Koja primitiva će biti selektovana u SPHIGS sceni, ako se pokaže na tačku gde se više primitiva preklapa u projekciji?

Napomene:

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

Rešenja zadataka

drugi kolokvijum 2012

1) Rešenje

```
/* RG 2011-2012: K2 */
#include <GL/glut.h>
#include <GL/GL.H>
#define BROJ_SEGMENATA 5
#define BROJ_ELEMENATA 5
void promenaProzora(int w, int h) {
    glViewport(0, 0, w, h);
}

void crtajStub(float *boja) {
    glPushMatrix();
    glPushAttrib(GL_ALL_ATTRIB_BITS);
    glColor3fv(boja);
    glBegin(GL_QUADS);
    glVertex3d( -0.1, 0, 0.02 );
    glVertex3d( 0.1, 0, 0.02 );
    glVertex3d( 0.1, 0.9, 0.02 );
    glVertex3d( -0.1, 0.9, 0.02 );
    glEnd();
    glBegin(GL_TRIANGLES);
    glVertex3d( -0.1, 0.9, 0.02 );
    glVertex3d( 0.1, 0.9, 0.02 );
    glVertex3d( 0, 1, 0.02 );
    glEnd();
    glPopAttrib();
    glPopMatrix();
}

void crtajSegmentOgrade(float *boja, int brElem) {
    glPushMatrix();
    glTranslated(0.15, 0, 0);
    for(int i = 0; i < brElem; i++) {
        crtajStub( boja );
        glTranslated(0.3, 0, 0);
    }
    glPopMatrix();

    glPushAttrib(GL_ALL_ATTRIB_BITS);
    glPushMatrix();
    glColor3f(0,0,0);
    glLineWidth(6);
    glTranslated(0,0.2,0.055);
    glBegin(GL_LINES);
    glVertex3d(0, 0, 0);
    glVertex3d(brElem*0.3, 0, 0);
    glEnd();

    glTranslated(0,0.6,0);
    glBegin(GL_LINES);
    glVertex3d(0, 0, 0);
    glVertex3d(brElem*0.3, 0, 0);
    glEnd();
    glPopMatrix();
    glPopAttrib();
}

void crtajOgradu(int brSeg, int vrata) {
    static float t_siva[]={0.3f,0.3f,0.3f};
    static float siva[]={0.5f,0.5f,0.5f};

    glPushMatrix();
    for(int i = 0; i < brSeg; i++) {
        if( i == vrata ) {
            glPushMatrix();
            glRotated(-60, 0, 1, 0);
        }
        if( i & 1 ) crtajSegmentOgrade(t_siva,BROJ_ELEMENATA);
        else crtajSegmentOgrade(siva, BROJ_ELEMENATA);
        if( i == vrata ) glPopMatrix();
        glTranslated(BROJ_ELEMENATA*0.3, 0, 0);
    }
    glPopMatrix();
}

void crtajScenu(void) {
    glClearColor(GL_COLOR_BUFFER_BIT |
GL_DEPTH_BUFFER_BIT);
    glLoadIdentity();
    gluLookAt(-12,10,15,4,0,0,1,0);
    crtajOgradu(BROJ_SEGMENATA, 2);
    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, GL_FILL);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-3, 3, -2, 2, 0, 30);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
}

int main(int argc, char **argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode (GLUT_RGB | GLUT_DEPTH);
    glutInitWindowSize(500, 500);
    glutInitWindowPosition(100, 100);
    glutCreateWindow("Ograda");
    init();
    glutDisplayFunc(crtajScenu);
    glutReshapeFunc(promenaProzora);
    glutMainLoop();
    return 0;
}
```

2) Rešenje

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

$$[X'' \ Y'' \ Z'' \ 1] = \begin{bmatrix} X' & Y' & Z' \\ W & W & W \end{bmatrix} 1]$$

$$M = \text{Ogledanje} * \text{Projekcija} * \text{Prozor}$$

$$\text{Ogledanje} = T(5, 0, 0) * R(0, 0, -\frac{\pi}{4}) * S(-1, 1, 1) * R(0, 0, \frac{\pi}{4}) * T(-5, 0, 0)$$

$$\text{Projekcija} = T(-2, 2, -2) * R(\frac{\pi}{4}, 0, 0) * Pp(2\sqrt{2})$$

$$\text{Prozor} = S(-1, -1) * S(\frac{0.4}{8}, \frac{0.3}{7}) * T(-0.2, -0.15) * R(\frac{\pi}{3}) * T(-0.2, -0.26)$$