

Računarska grafika

OpenGL 2

Naš prvi OpenGL program

```
#include <GL/glut.h>
```

```
int main(int argc, char** argv) {  
    glutInit(&argc, argv);  
    glutInitWindowPosition(100, 100);  
    glutInitWindowSize(400, 300);  
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);  
    glutCreateWindow("Moj prozor");  
    glutMainLoop();  
    return 0;  
}
```

Primjer 1

glShadeModel

- Podešava način na koji će poligon biti obojen (ispunjen)

```
void glShadeModel(GLenum mode);
```

- mode parametar može biti postavljen na jednu od dvije konstante (GL_FLAT ili GL_SMOOTH)
- *default* način crtanja je GL_SMOOTH
- Ukoliko se za crtanje koristi GL_FLAT boja poligona se bira na osnovu vrste poligona

Primjer 2

glutMouseFunc

- Postavlja *callback* funkciju za miš

```
void glutMouseFunc(void (*func)(int button, int state, int x, int y));
```

- *x* i *y* parametri predstavljaju lokaciju miša u trenutku okidanja *callback* funkcije
- *button* parametar označava taster koji je uzrokovao pozivanje *callback* funkcije (GLUT_LEFT_BUTTON, GLUT_MIDDLE_BUTTON, GLUT_RIGHT_BUTTON)
- *state* parametar daje informaciju o tome je li taster pritisnut ili otpušten (GLUT_UP, GLUT_DOWN)

glutMotionFunc/glutPassiveMotionFunc

- Postavljaju *callback* funkcije koje se vežu za pomjeranje pozicije miša
 - Kada je taster miša pritisnut

```
void glutMotionFunc(void (*func)(int x, int y));
```

- Kada taster miša nije pritisnut

```
void glutPassiveMotionFunc(void (*func)(int x, int y));
```

Primjer 3

glutPostRedisplay (*pseudo*)

```
int do_display = 0;

void glutPostRedisplay(void) {
    do_display = 1;
}

void glutMainLoop(void) {
    for(;;) {
        event ev = get_OS_event();
        switch(ev.type) {
            case keyboard_event: call_keyboard_function(ev); return;
            case mouse_event:    call_mouse_function(ev); return;
            /* ... */
        }
        if( do_display ) {
            call_display_function();
            do_display = 0;
        } else {
            call_idle_function();
        }
    }
}
```

glutKeyboardFunc

- Postavlja *callback* funkciju za unos sa tastature

```
void glutKeyboardFunc(void (*func)(unsigned char key, int x,  
int y));
```

- key parametar predstavlja ASCII vrijednost unesenog karaktera
- x i y parametri predstavljaju lokaciju miša u trenutku okidanja *callback* funkcije

glutSpecialFunc

- Postavlja *callback* funkciju za unos funkcijskih i navigacijskih tipki

```
void glutSpecialFunc(void (*func)(int key, int x, int y));
```

- x i y parametri predstavljaju lokaciju miša u trenutku okidanja *callback* funkcije
- key parametar nosi vrijednost jedne od GLUT_KEY_* konstanti (GLUT_KEY_F1, GLUT_KEY_LEFT, GLUT_KEY_HOME ...)

Primjer 4

glutReshapeFunc

- Postavlja *callback* funkciju za tretiranje promjene veličine prozora

```
void glutReshapeFunc(void (*func)(int width, int height));
```

- `width` i `height` parametri predstavljaju nove dimenzije prozora
- funkcija će se pozvati i u slučaju inicijalnog prikazivanja prozora
- *default callback* će izvršiti poziv sljedeće funkcije:

```
glViewport(0, 0, width, height);
```

Geometrijske transformacije

- Translacija

```
void glTranslatef(GLfloat x, GLfloat y, GLfloat z);
```

- Skaliranje

```
void glScalef(GLfloat x, GLfloat y, GLfloat z);
```

- Rotacija

```
void glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat z);
```

Geometrijske transformacije

- Translacija

`void glTranslatef(GLfloat x, GLfloat y, GLfloat z);` $2D \rightarrow z = 0$

- Skaliranje

`void glScalef(GLfloat x, GLfloat y, GLfloat z);` $2D \rightarrow z = 1$

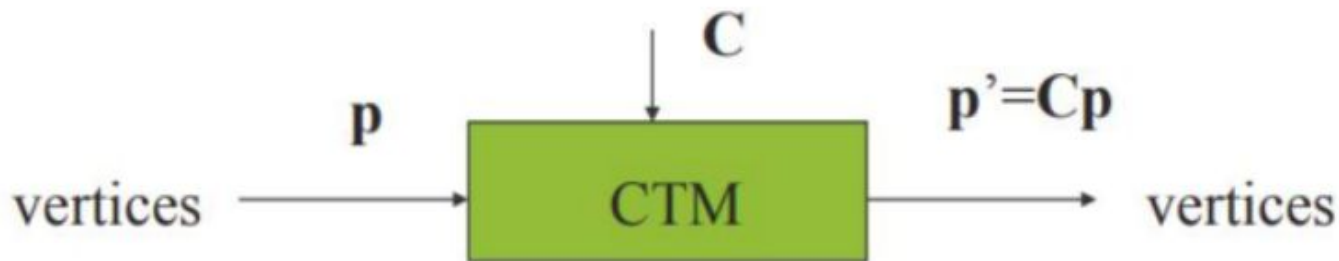
- Rotacija

`void glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat z);`

$2D \rightarrow x = y = 0, z = 1$

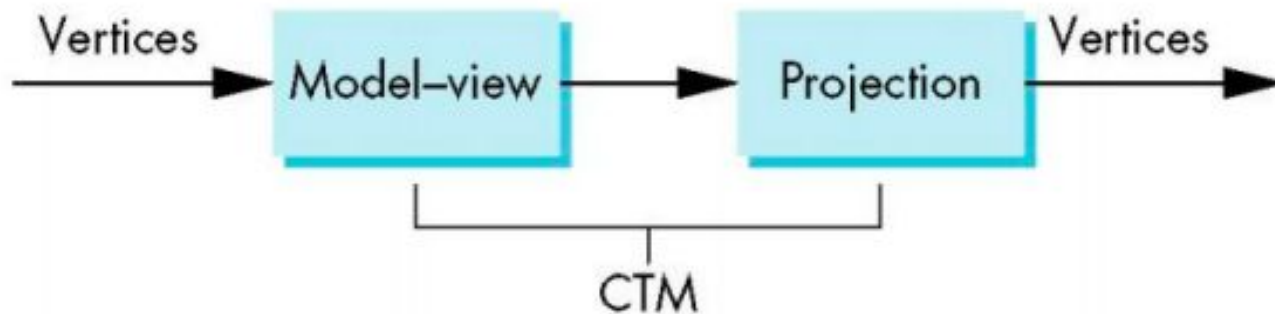
OpenGL matrice

- U bilo kojem trenutku postoji trenutna matrica koja se primjenjuje na tačke na sceni.



OpenGL matrice

- Model-view - geometrijske transformacije i transformacije pogleda (`glTranslate`, `glRotate`, `glScale`, `gluLookAt...`)
- Projection - projekcija (`glViewport`, `gluPerspective`, `glOrtho`, `glFrustum...`)



glMatrixMode

- Postavlja matricu za predstojeće operacije

```
void glMatrixMode(GLenum mode);
```

- mode ima jednu od dvije vrijednosti
 - GL_MODELVIEW
 - GL_PROJECTION

glMatrixMode

- Postavlja matricu za predstojeće operacije

```
void glMatrixMode(GLenum mode);
```

- mode ima jednu od dvije vrijednosti
 - GL_MODELVIEW
 - GL_PROJECTION

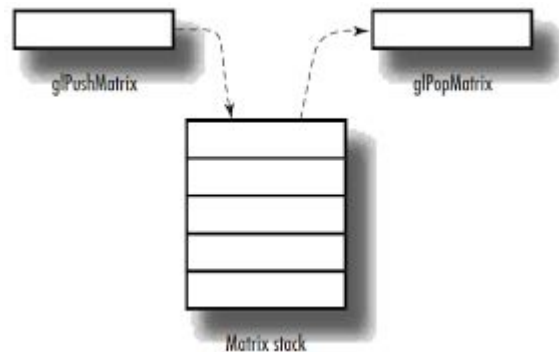
glLoadIdentity

- Postavlja jediničnu matricu za trenutnu

```
void glLoadIdentity();
```

Matrični *stack*

- Poziv `glPushMatrix()`, povećava *stack pointer*, a trenutna matrica se kopira na vrh *stack-a*
- Bilo koja transformacija (`glRotate`, `glTranslate...`) djelovat će samo na matricu na vrhu *stack-a*.
- `glPopMatrix()` uništava matricu na vrhu *stack-a* te druga matrica na postaje trenutna.



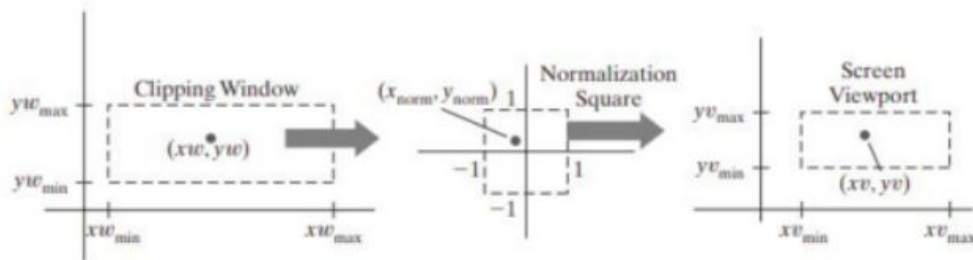
Primjer 5

gluOrtho2D

- Definiše 2D *clipping* prozora

```
void gluOrtho2D(GLdouble xwmin, GLdouble xwmax, GLdouble  
ywmin, GLdouble ywmax);
```

- *Default* koordinate [-1.0, 1.0]



Primjer 6

Primjer 7

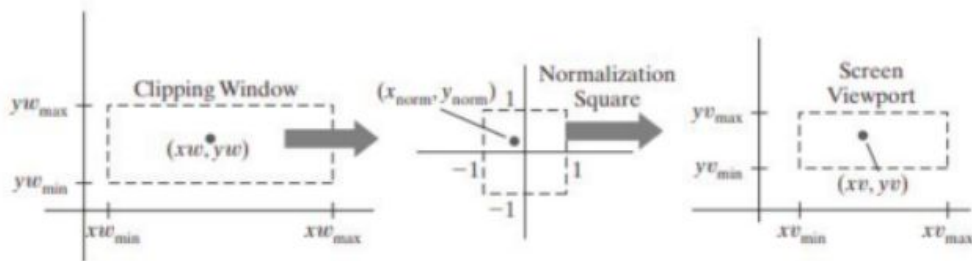
Primjer 8

glViewport

- Specificira granice *viewport*-a pomoću pozicije donjeg lijevog ugla *viewport*-a i njegove širine i visine u pikselima

```
void glViewport(GLint xvmin, GLint yvmin, GLsizei vpWidth,  
GLsizei vpHeight);
```

- *Default* vrijednosti za navedene parametre jednake su poziciji i veličini aktivnog prozora.



Primjer 9

Hvala na pažnji!