




Računarska grafika

Biblioteka OpenGL

Godina 2017/2018.

Ivan Čukić

ivan@math.rs
poincare.math.rs/~ivan
Matematički fakultet, Univerzitet u Beogradu



Uz veliku zahvalnost
Aleksandru B. Samardžiću

Legenda



OpenGL



OpenGL Core Profile



OpenGL Compatibility Profile



OpenGL Utility Library



OpenGL Utility Toolkit



OpenGL Extension Wrangler



Sekcija u beleškama za predavanja, Predrag Janičić



OpenGL Programming Guide (Red Book), 7th edition



Link ka objašnjenju pojma na Wikipedia.org



Link ka arhivi sa izvornim kôdom

UVOD

OpenGL





GLUT

Osnove iscrtavanja

Mogućnosti

- Definisanje tačaka, linija, poligona u prostoru
- Postavljanje osvetljenja
- Postavljanje pozicije posmatrača i osobina pogleda

Sintaksa

- Komande imaju prefix `gl`
`glBegin`  , `glVertex`  
- Konstante imaju prefix `GL_`
`GL_MULTISAMPLE`, `GL_POLYGON`
- Tipovi podataka imaju prefix `GL`
`GLbyte`, `GLdouble`

 Red Book: **1.2**

Osobine

- Minimalan API
- Ne zavisi od platforme
- Ne zavisi od UI biblioteke
- Mašina stanja



 State machine 

Overloading

`glCommand n t t [v]`

- n - koliko funkcija prima argumenata
- t - oznaka tipa argumenta
- v - ako se argumenti prosleđuju kao vektor, odnosno pokazivač na nulti element

Primeri:

- `glVertex3f(x, y, z)` GL COMPAT 
- `glVertex2iv(niz)` GL COMPAT 



Tipovi

sufiks	tip podataka	OpenGL tip
b	8-bit integer	GLbyte
s	16-bit integer	GLshort
i	32-bit integer	GLint
f	32-bit real	GLfloat
d	64-bit real	GLdouble
ub	8-bit unsigned integer	GLubyte, GLboolean
us	16-bit unsigned integer	GLushort
ui	32-bit unsigned integer	GLuint, GLenum, GLbitfield

Pipeline

OpenGL 1.x

Fixed pipeline

- transformacija temena  8
- odsecanje (clipping)  5.2
- rasterizacija
- osvetljavanje
- operacije nad pikselima



Compatibility profile in 3.2

`glBegin`  

 Red Book: **1.5**

OpenGL 2.x, 3.x

Programmable pipeline

- Svaki korak je moguće programirati  Shader 
- Vertex shader
- Geometry shader
- Fragment (pixel) shader

Core profile in 3.2



`glEnable`  

Pomočné biblioteky

- GLU - OpenGL Utility Library
`gluOrtho2D` GLU 
- GLUT - OpenGL Utility Toolkit
`glutMainLoop` GLUT 
- GLEW - OpenGL Extension Wrangler Library
`glewGetContext` GLEW

GLUT biblioteka

Mogućnosti:

- Korisničko okruženje  2.3  3.1
- Obrada događaja
- Pomoć pri iscrtavanju

Alternative: Qt, GTK+, Swing, ...




 Program: **01-quad** 

GLUT biblioteka

Inicijalizacija biblioteke:

- `glutInit` GLUT 
- `glutInitDisplayMode` GLUT 

Kreiranje prozora:

- `glutInitWindowSize` GLUT 
- `glutInitWindowPosition` GLUT 
- `glutCreateWindow` GLUT 

 Program: **01-quad** 

GLUT biblioteka

Callback funkcije:

- `glutDisplayFunc` [GLUT](#) 

UI petlja i iscrtavanje:

- `glutMainLoop` [GLUT](#) 

- `glutSwapBuffers` [GLUT](#) 

 Program: **01-quad** 

Osnovne komande

Inicijalizacija OpenGL-a:




- `glClearColor` [GL CORE](#) , `glClear` [GL CORE](#) 
- `glEnable` [GL CORE](#) , `glDisable` [GL CORE](#) 

Osnove iscrtavanja:

- `glBegin` [GL COMPAT](#) , `glEnd` [GL COMPAT](#) 
- `glColor` [GL COMPAT](#) 
- `glVertex` [GL COMPAT](#) 

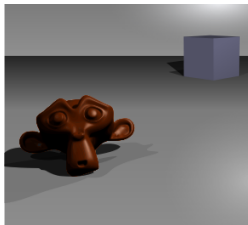
 Program: **01-quad**   Red Book: **2.1**

Parametri iscrtavanja

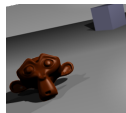
- Model boja: RGB, RGBA  12.4.3
- Double-buffering  3.3.4
- Z-buffer  11.4

Double-buffering

Ekran

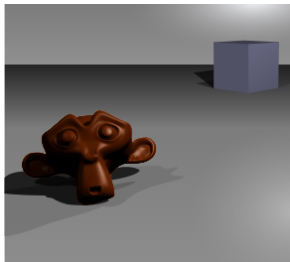


Prikazani bafer

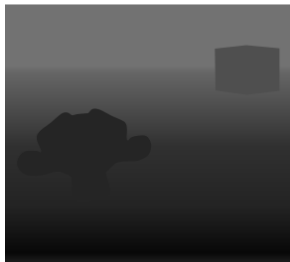


Bafer po kome crtamo

Z-buffer




Iscrtana slika



Z-buffer

📦 Program: **02-quad-test-depth** [↗](#)

Podrazumevani pogled

- Vidljivi prostor $[-1, 1]^3 \subset \mathbb{R}^3$
- Ortogonalna projekcija na xy ravan (front-elevation)
 8.3

DODATAK

Editori

Kompajler

make

Editor

- Emacs
- Vim
- Kate, KWrite, etc.

Kompajler

Kompilacija:

```
gcc izvorni_fajl.c -o izvrsni_fajl
```

Pokretanje

```
./izvrsni_fajl
```

Linkovanje dodatnih biblioteka

-l - library

```
gcc ... -lbiblioteka1 -lbiblioteka2 ...
```

```
gcc ... -lm -lGL -lGLU -lglut ...
```

-L - library-path

```
gcc ... -L/usr/X11R6/lib -L/usr/pkg/lib ...
```

Include putanje

-I - include-path

```
gcc ... -I/usr/X11R6/include -I/usr/pkg/include ...
```


Drugi korisni argumenti

- Wall – prijavljuje sva upozorenja
- ansi – prati ANSI C standard
- g – uključuje debug simbole u izvršni fajl

Makefile, GNU make

Makefile pravilo:

```
    cilj : zavisnosti  
    ->      komande za pravljenje cilja
```

Primer 1. Kompilacija programa

```
    program : main.c  
    ->      gcc -o program -lm -ansi -g -Wall main.c
```

Primer 2. Brisanje binarnih fajlova

```
    clean :  
    ->      -rm -f program  
    ->      -rm -f *.o
```

-> je tabulator. Ako se zameni razmacima, komanda make neće raditi