CSE328 Fundamentals of Computer Graphics

OpenGL Programming: A Quick Walkthrough (1)

Yicheng Lin

Dept. of Computer Science, Stony Brook University

yiclin@cs.stonybrook.edu

Outline

- TA & Homework info
- Introduction to OpenGL
- Programming Environment: VS+GLUT
- OpenGL 2D: Examples & Explanation
- Some Tips

TA & Homework Info

- Email: yiclin@cs.stonybrook.edu
- Office hours: TuTh 1pm-2:30pm, NCS Bldg 132
  Please email before visit, thanks!
- Course TA webpage:
  http://www3.cs.stonybrook.edu/~yiclin/cse328/index.html

TA & Homework Info

- Please compress your homework in a zip file and send to my email
- **File name: hw1_yourname.zip**
- Please include:
  - **Your C/C++ code** (SOURCE CODE ONLY)
  - **A release version of your program** (Windows executable)
  - **A Readme file**

Introduction to OpenGL

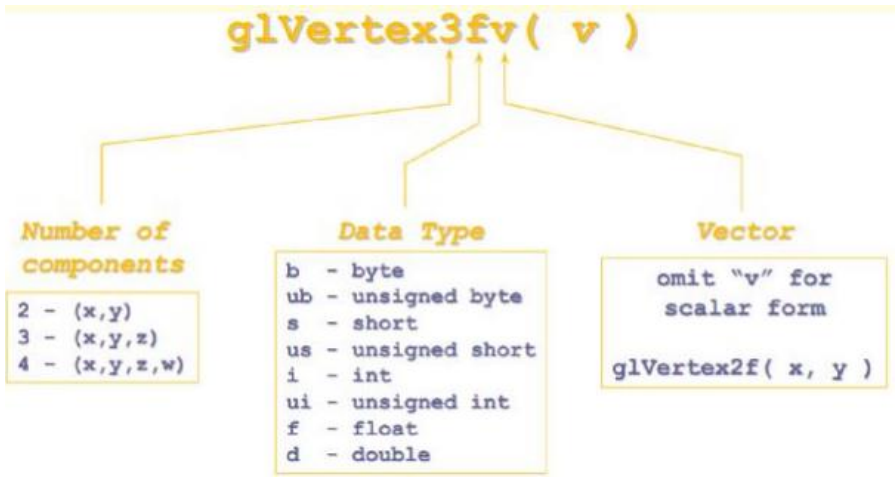
Open Graphics **L**ibrary is

- A cross-platform, industry standard library of computer graphics
- A set of graphics APIs for "drawing something" and "setting something"
- A state machine that use commands to control different states and keep them until later changed

Introduction to OpenGL

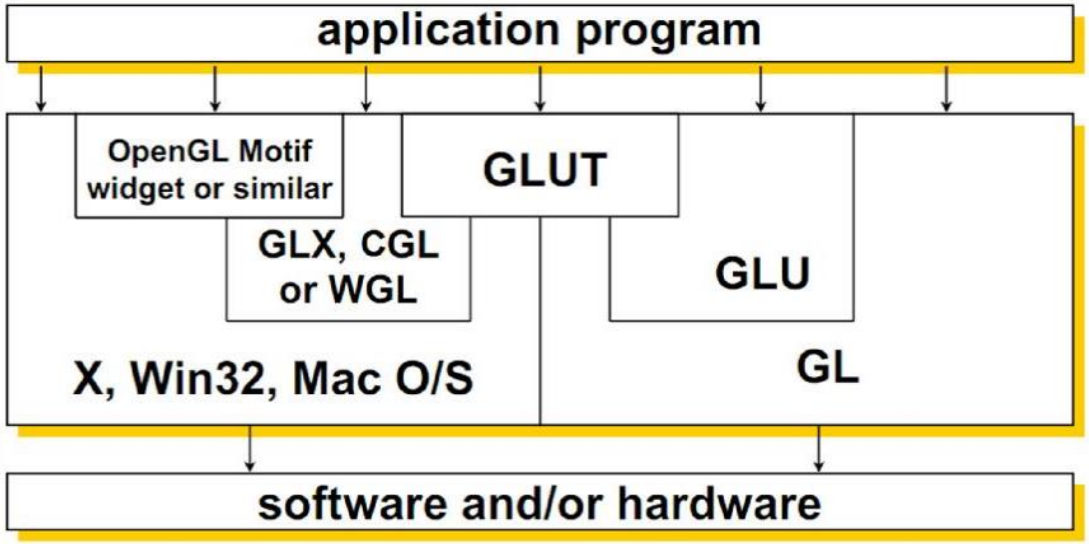
OpenGL language naming convention

- Defined constants begin with **GL_**
  - e.g., GL_TRIANGLES, GL_CURRENT_COLOR
- OpenGL commands are prefixed by **gl**
- OpenGL commands may have postfixes indicating number and data type of arguments



Introduction to OpenGL

- OpenGL is focused on drawing and has no concept of windowing and I/O



Programming Environment

- **Recommended: MS Visual Studio 2010 + GLUT**
- Easy to use and debug

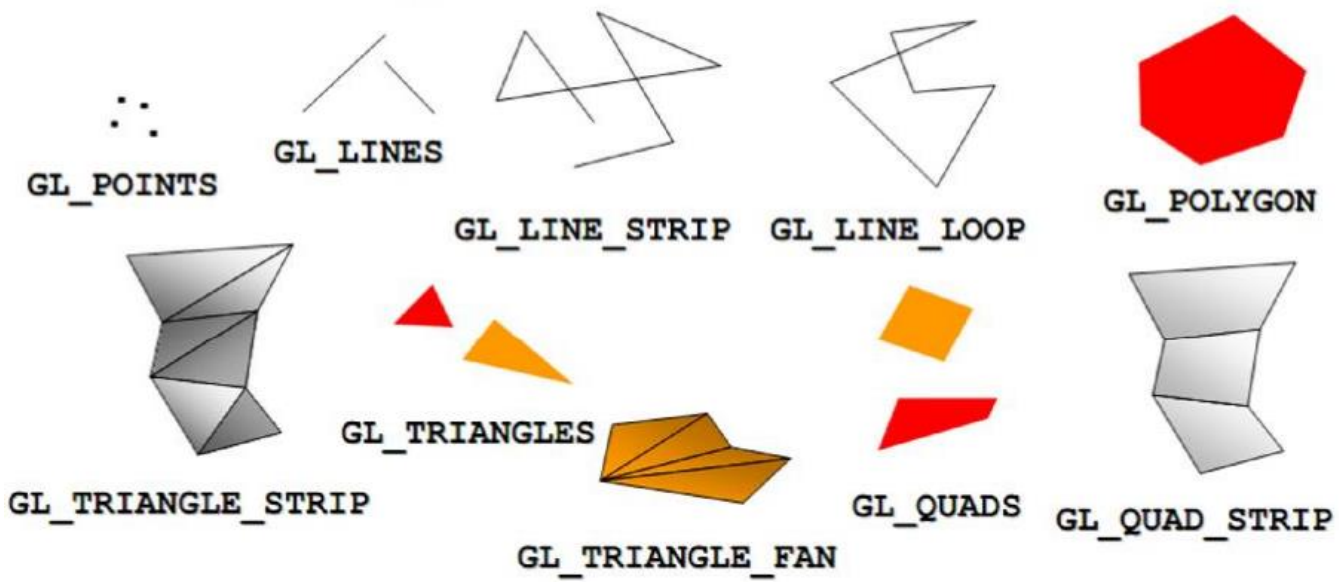
- Downloads
- https://www.dreamspark.com/student/Default.aspx
- http://user.xmission.com/~nate/glut.html

Example 1

- Initialize the GLUT window
- Draw a rectangle

Example 2

- Use different geometric primitives
- Change colors, line width, point size etc.
- Handle keyboard events



Example 3

- 2D transformation
- Handle mouse events

Example 4

- Maintain the list of shapes (C++)
- Visual aid design

Some Tips

- Start early
- Consider the structure of your code before you start
- Code and test bit by bit
- Write good comments
- **Try googling things you don’t know**

Q&A

yiclin@cs.stonybrook.edu

http://www3.cs.stonybrook.edu/~yiclin/cse328/index.html