

COMP 360/560

Tutorial

Jan 14, 2015 SYM II
Lab0

Today we will ...

- Familiarize everyone with
 - Programming environment
 - FLTK, Code structure
 - Labs and submission
 - OpenGL: some 2D and 3D exercises

❖ Programming

environment

Visual C++ 2010

Express

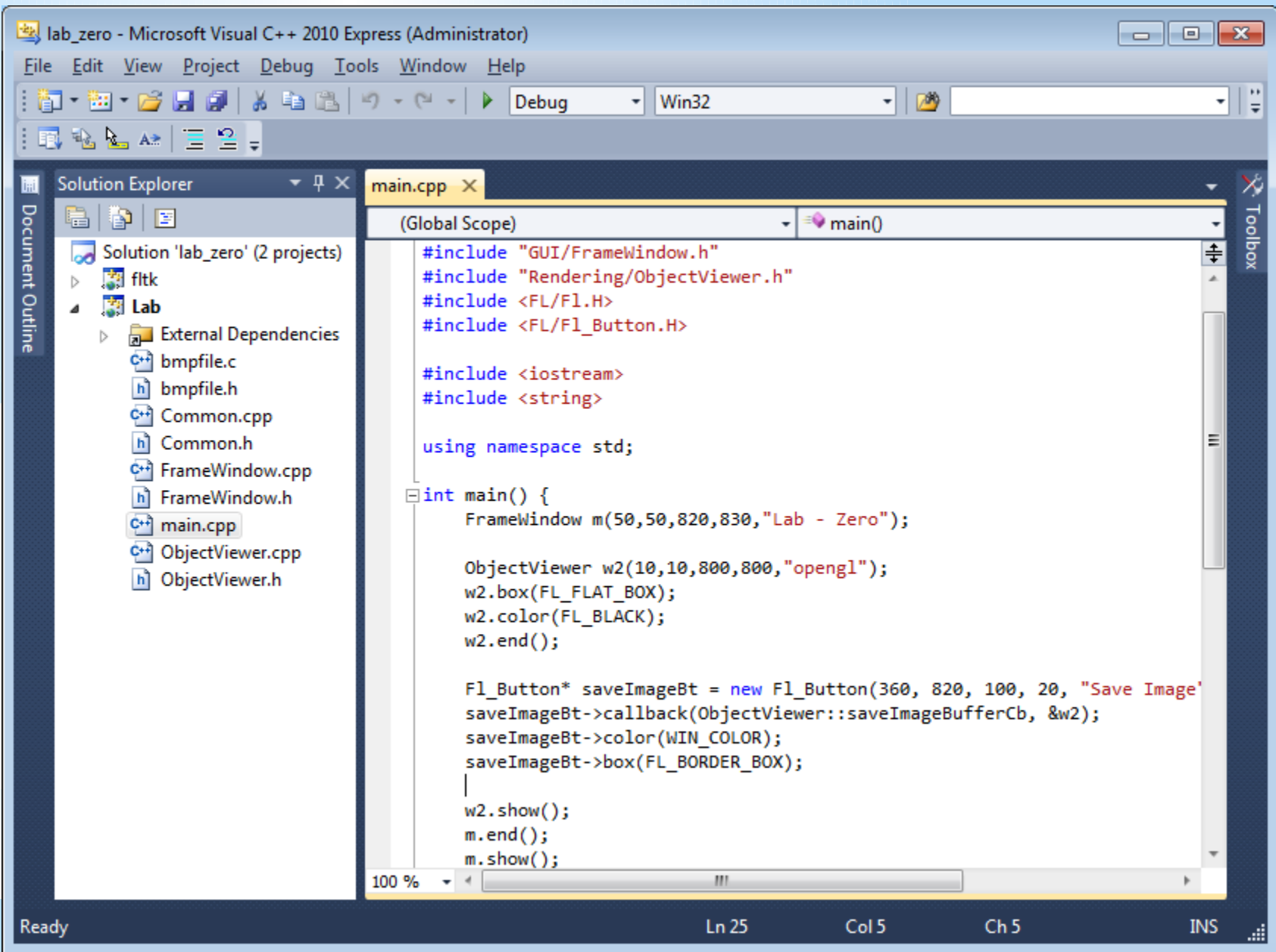
All template code will be given in Visual C++ 2010 Express

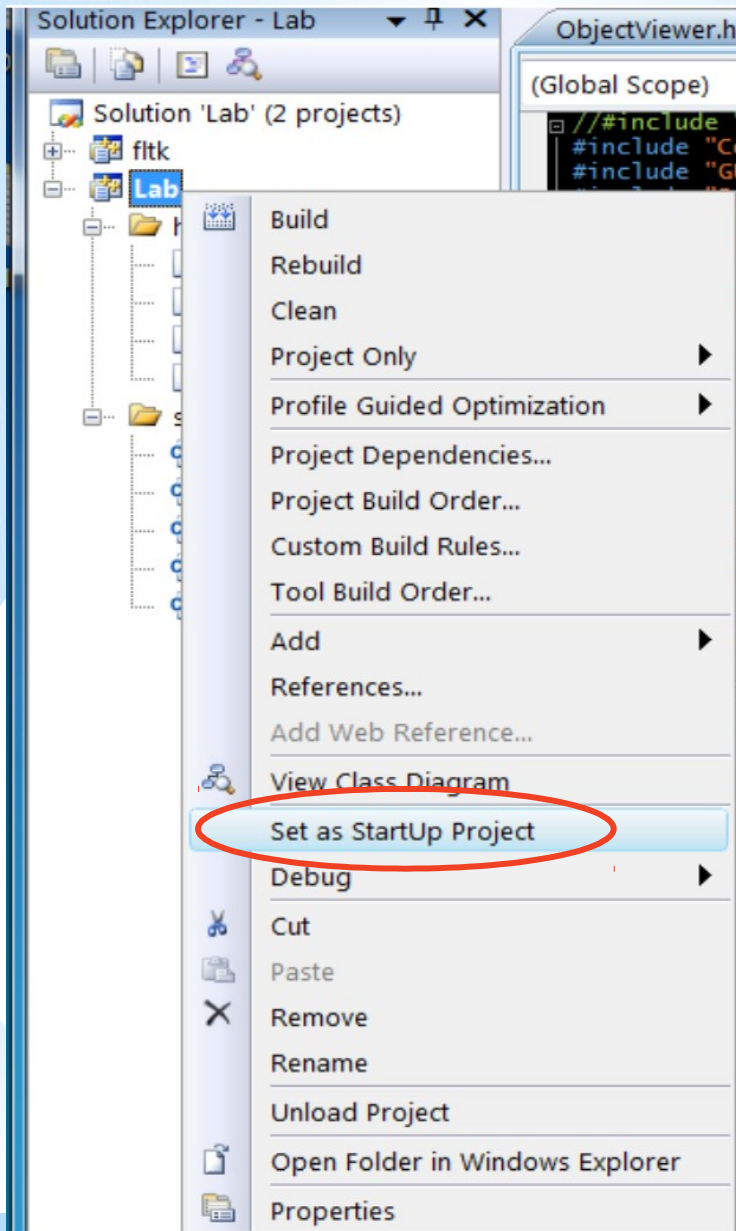
Download VC++ 2010 Express:

<http://www.microsoft.com/visualstudio/eng/products/visual-studio-2010-express>

Intro to C++:

<http://www.cplusplus.com/files/tutorial.pdf>





- Set *Lab* as the **Startup Project**
- Try to compile project with **F5**, or the **green arrow** button
- Debug Mode, slower but has debugger
- Release Mode, faster, when your code is stable

lab_zero (Debugging) - Microsoft Visual C++ 2010 Express (Administrator)

File Edit View Project Debug Tools Window Help

Process: [1676] Lab.exe Thread: [2112] Main Thread

Solution Explorer

- Solution 'lab_zero' (2 projects)
- ftk
- Lab
 - External Dependencies
 - bmpfile.c
 - bmpfile.h
 - Common.cpp
 - Common.h
 - FrameWindow.cpp
 - FrameWindow.h
 - main.cpp
 - ObjectViewer.cpp
 - ObjectViewer.h

main.cpp

```

(Global Scope)
main()
#include "GUI/FrameWindow.h"
#include "Rendering/ObjectViewer.h"
#include <FL/Fl.H>
#include <FL/Fl_Button.H>

#include <iostream>
#include <string>

using namespace std;

int main() {
    FrameWindow m(50,50,820,830,"Lab - Zero");

    ObjectViewer w2(10,10,800,800,"opengl");
    w2.box(FL_FLAT_BOX);
    w2.color(FL_BLACK);
    w2.end();
  
```

Autos

Name	Value	Type
m	{_triButton=0x5aa9a140 }	FrameW
w2	{_w=18 _h=1638048 _rot=6.489 }	ObjectVi

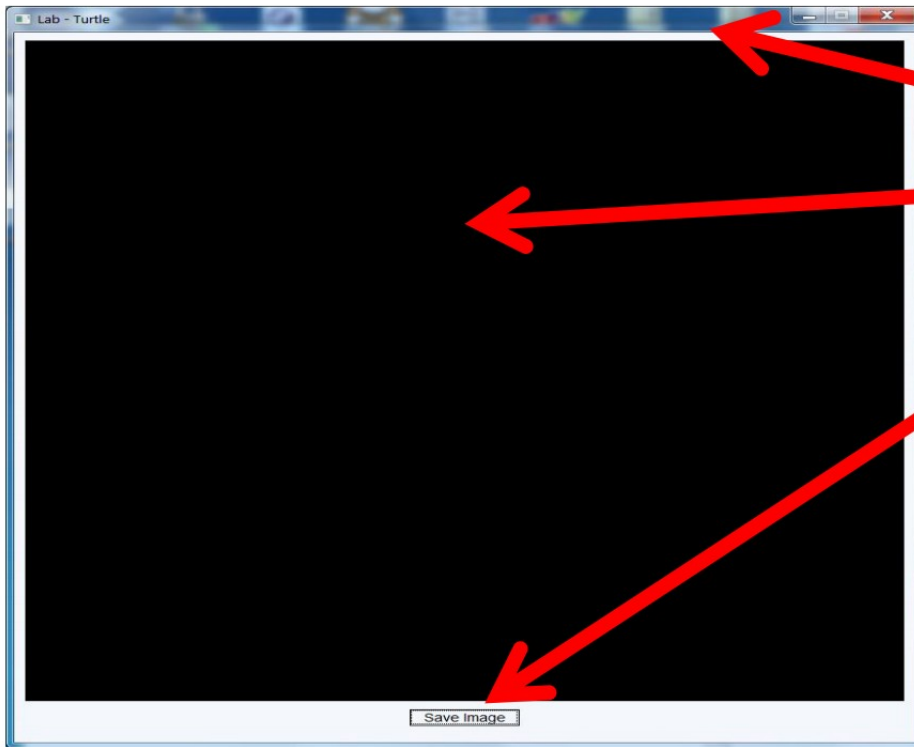
Call Stack

Name	Lang
Lab.exe!main() Line 16	C++
Lab.exe!WinMain(HINSTANCE__ * hInstance, HINS	C
Lab.exe!_tmainCRTStartup() Line 547 + 0x2c byte	C
Lab.exe!WinMainCRTStartup() Line 371	C
kernel32.dll!751633aa()	
[Frames below may be incorrect and/or missing, r	
ntdll.dll!777c9f72()	
ntdll.dll!777c9f45()	

Ready Ln 16 Col 1 Ch 1 INS

❖ Code Structure

main.cpp



```
FrameWindow m(50,50,820,830,"Lab - Tu  
ObjectViewer w2(10,10,800,800,"opengl  
w2.box(FL_FLAT_BOX);  
w2.color(FL_BLACK); //w2.position(600  
w2.end();  
  
Fl_Button* saveImageBt = new Fl_Butto  
saveImageBt->callback(ObjectViewer::s  
saveImageBt->color(WIN_COLOR);  
saveImageBt->box(FL_BORDER_BOX);  
  
w2.show();  
m.end();  
m.show();  
return Fl::run();
```

ObjectViewer.cpp

- OpenGL code
- We will be looking at
 - 2D drawing
 - 3D drawing (2d is just a subspace/subset/plane/projection).
 - Simple event handling

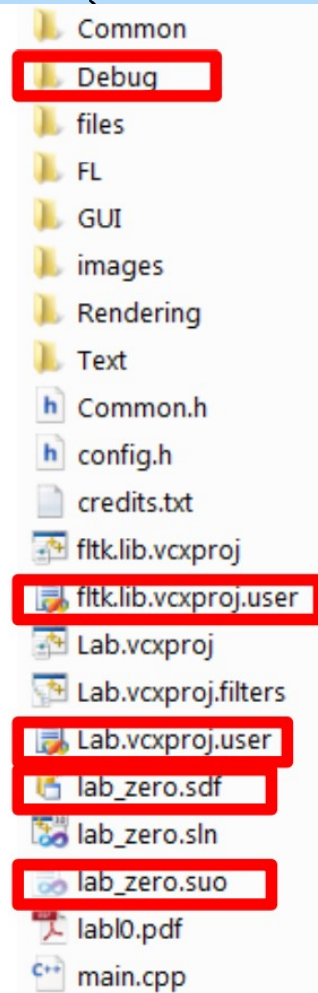
❖ Labs and submission

title: labX_yourNetID. (lab0_by8.zip)

Submit to owlspace. (Due: 2015/01/19 11:55)

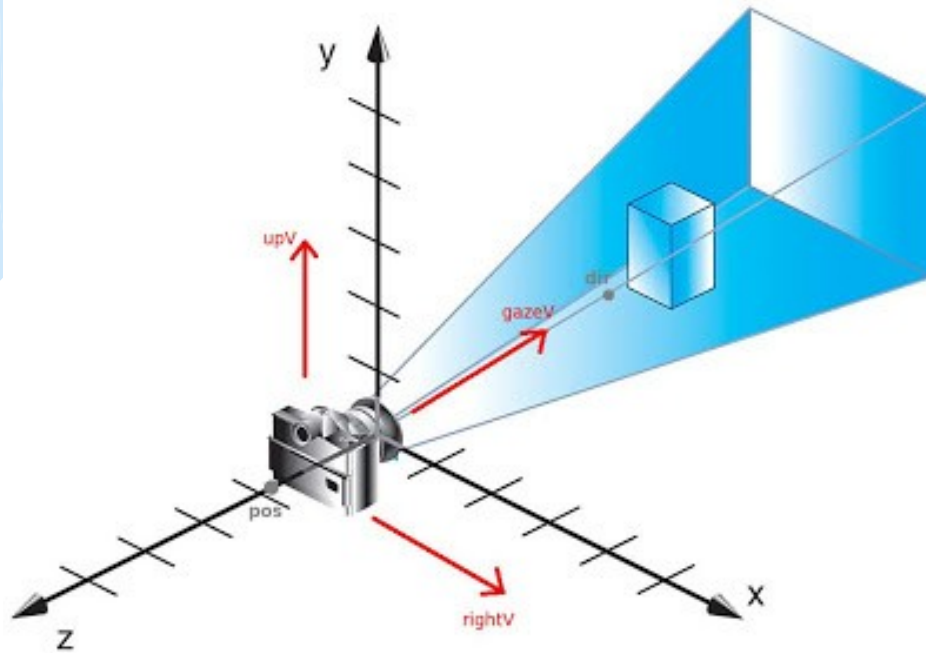
- Add a README with your name
- Always delete the binaries or unnecessary files to keep the file size low for email submissions

Always test compile and run before Submission !



❖ Exercise 2D and 3D

drawing



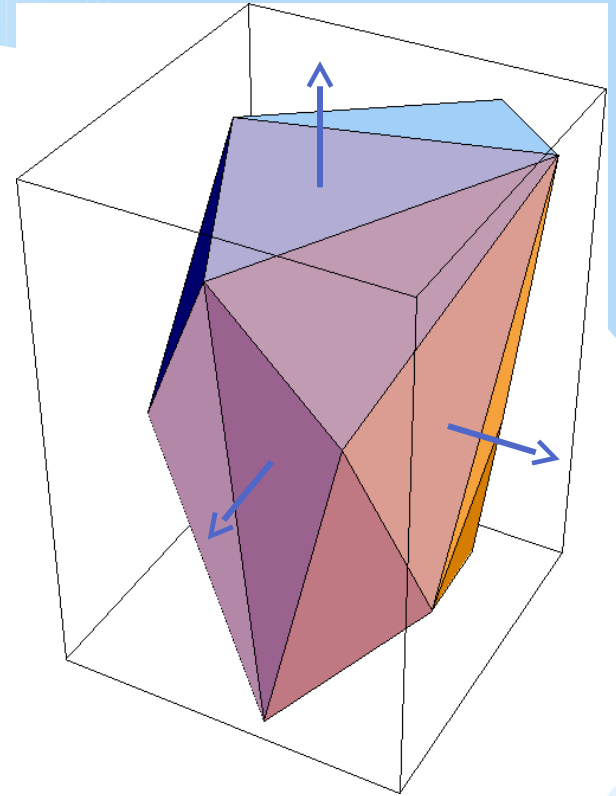
OpenGL resources:

<http://www.glprogramming.com/red/>

<http://nehe.gamedev.net/>

Exercise 1

- Translate the triangle so that its lengths are preserved, but one of its vertices correspond to the point of the mouse click.
- Hint: use `getWidth()` and `getHeight()`



Exercise 2

- Continuously rotate the triangle about one of its vertices
- Hint: draw() function is being repeatedly called
- There's already a variable call rot

Exercise 3 and 4

- Submit before **11:59PM next Monday** (Sept. 2, 2013).
- 3D mode: Rotate the cube when the key 'r' is held down.
- 2D mode: When the mouse is clicked in 2D mode, place a triangle down such that the mouse point coincides with the vertex of the triangle.

Done! Have fun!