Introduction to Computer Graphics
Practical Session – OpenGL Framework

In the first practical session you will setup the C++ framework used throughout the course (exercises, practice sessions and project). To confirm the success in setting up the framework, **each student has to submit a snapshot of the rendering result after running the code.** Failure in doing so results in 5% less of the total exercise and project grade. Note that, except for valid reasons, **we won't assist you in setting up the framework once the submission deadline is over.**

Supported system. We only support the Linux operating system (with GNU make g++) installed in *the lab machines*. Other systems are not supported and the instructions are provided as is. If you have trouble setting up the framework on your personal laptop, it might be helpful to check Sec. 4. Note that your laptop might be inadequate to run the exercises if your graphic card doesn't support OpenGL above version 3.3. If this is the case, please use the lab machines.

QtCreator. We recommend this the IDE for C++/OpenGL programming. This IDE supports C++, GLSL as well as the CMake build system. To configure your project simply open the *CMakeLists.txt* file within QtCreator. We also recommend to do this **after** you compiled by command line following the instructions below.

"My files" folder. In the lab machines, the code won't compile in the "my files" folder. In addition, every file in your home directory is removed once you log out from the computer. Thus, make sure to save your work.

1 Ubuntu 12.04

1. The framework dependencies (`cmake`, `glfw` and `glew`) are already installed on the lab machines. If you need to install them on your personal machine, follow these instructions:

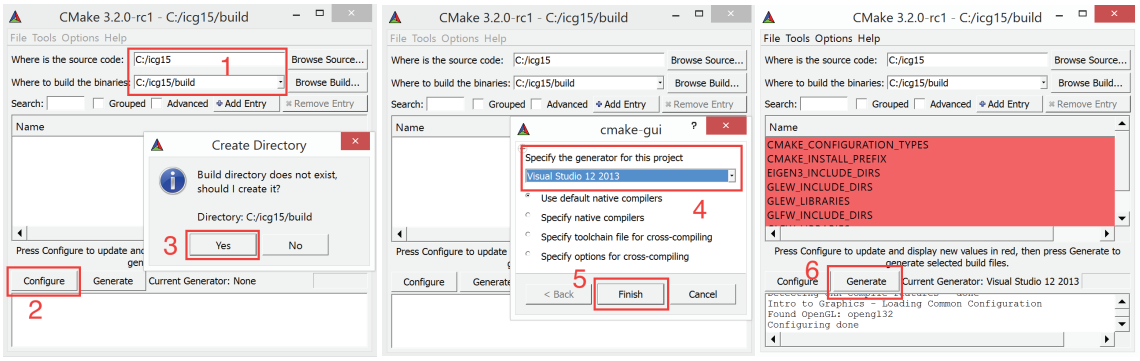
```
~/: sudo apt-get install cmake
~/: sudo apt-get install libglew libglew-dev
~/: sudo apt-get install libglfw2 libglfw-dev
```

2. Download/configure/compile your code

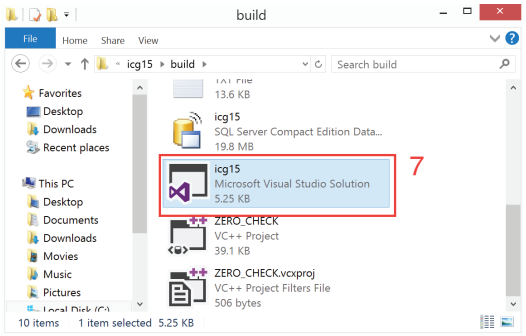
```
~/: git clone https://git.epfl.ch/repo/icg15.git
~/: cd icg15
~/ icg15: mkdir build
~/ icg15: cd build
~/ icg15/build: cmake ../                    # creates a Makefile
~/ icg15/build: make                         # builds the solution
~/ icg15/build: cd helloworld
~/ icg15/build/helloworld: ./helloworld      # executes the application
```

2 Windows 7/8, Visual Studio 12 (2013), 32-bit platform

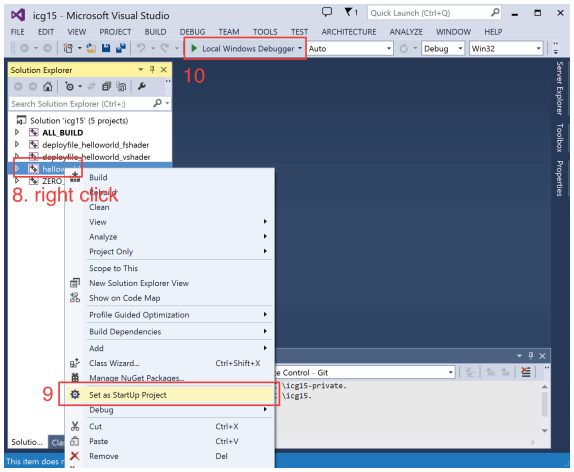
1. Download and install Visual Studio from *http://dreamspark.epfl.ch*
2. Download and install CMake from *http://http://www.cmake.org*
3. Install git and a git client (e.g GitHub Windows)
4. Clone the software framework from *https://git.epfl.ch/repo/icg15.git*
5. We provide the binaries for `glfw` and `glew`. If you need to build them yourselves, refer to Sec. 4.
6. Follow the snapshot tutorial in the figure below.



Use CMake (minimum version 2.8) to configure and generate the Visual Studio solution



Open the *icg15.sln* file in your *build* folder



Build all (F7) then select the project to be run by right clicking on a project and choose *Set as StartUp Project*. Finally run your application (F5)

3 Mac OSX 10.10.2

1. Install XCode (from the Apple store) and its *command line tools*
2. Install homebrew from `http://brew.sh`
3. Install the necessary libraries:

```
~/: brew install cmake                      # tested with cmake-2.8.12.2
~/: brew install homebrew/versions/glfw2    # tested with glfw2-2.7.9
~/: brew install glew                       # tested with glew-1.10.0
```

4. Download/configure/compile your code

```
~/: git clone https://git.epfl.ch/repo/icg15.git
~/: cd icg15
~/ icg15: mkdir build
~/ icg15: cd build
~/ icg15/build: cmake ../                   # creates a Makefile
~/ icg15/build: make                        # builds the solution
~/ icg15/build: cd helloworld
~/ icg15/build/helloworld: ./helloworld     # executes the application
```

4 Troubleshooting

Following are a few notes we collected from the previous year's course. If you have other tips to set up the framework, please share with your classmates via Moodle.

4.1 OpenGL version

The framework requires OpenGL newer than version 3.3. To check the OpenGL version in your laptop, follow these guides for Windows, Mac Os and Linux. If your OpenGL is outdated, you can update the graphic card driver.

Windows 7, Intel HD4000 graphic card. We noticed on one laptop with an Intel HD4000 graphic card the framework runs without problems only after **updating the card drivers**.

Windows 7, dual graphic card. On laptops having a dual graphic card, start visual studio by right clicking on the application and "Run using High Performance Graphics". Otherwise, GLFW will fail to open an OpenGL ≥ 3.3 context.

4.2 Windows external libraries.

If you encounter linking errors, you might need to build `glfw` and `glew` on your own. Download the source files from GLFW2 and GLEW, build, and copy the binaries to the folder: `icg15/external/{glfw,glew}/lib`.

4.3 Folder name

Be careful not to use folder names containing **unicode** characters (french accents, etc ...). This is not advisable when you do software development as it might give you weird errors. We also generally advise not to use **spaces** in file/folder names.