Prof. Dr. Leif Kobbelt
Dr. Jan Möbius,
Patrick Schmidt, Moritz Ibing

Basic Techniques in Computer Graphics
Assignment 1

Date Published: October 20th 2021,   Date Due: October 27th 2021

- All assignments (programming and theory) have to be completed in teams of 3–4 students. Teams with fewer than 3 or more than 4 students will receive no points.

- Hand in **one solution per team per assignment**.

- Every team must work independently. Teams with identical solutions will receive no points.

- Solutions are due 14:15 on October 27th 2021 via Moodle. Late submissions will receive zero points. No exceptions!

- Instructions for **programming assignments**:

  - Make sure you are part of a Moodle group with 3-4 members. See "Group Management" in the Moodle course room.

  - Download the solution template (a zip archive) through the Moodle course room.

  - Unzip the archive and populate the `assignmentXX/MEMBERS.txt` file. The names and student ids listed in this file **must match** your moodle group **exactly**.

  - Complete the solution.

  - Prepare a new zip archive containing your solution. It must contain exactly the files that you changed. **Only change the files you are explicitly asked to change in the task description.** The directory layout must be the same as in the archive you downloaded. (At the very least it must contain the `assignmentXX/MEMBERS.txt`.)

  - One team member uploads the zip archive through Moodle before the deadline, using the group submission feature.

  - Your solution must compile and run correctly **on our lab computers** by only inserting your **assignment.cc** and **shader files** into the Project. If it does not compile on our machines, you will receive no points. If in doubt you can test compilation in the virtual machine provided on our website.

- Instructions for **text assignments**:

  - Prepare your solution as a single pdf file per group. Submissions on paper will not be accepted.

  - If you write your solution by hand, write neatly! Anything we cannot decipher will receive zero points. No exceptions!

  - Add the names and student ID numbers of all team members to every pdf.

  - Unless explicitly asked otherwise, always justify your answer.

  - Be concise!

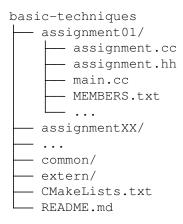 - Submit your solution via Moodle, together with your coding submission.

Exercise 1   Preparation [20 Points]

Make yourself familiar with the framework we are going to use for the practical assignments.

(a)   Download and Unpack

Go to the Moodle course room (Assignment 1), download `basic-techniques.zip` and unpack it. The contents of this archive will be required for this and all future programming assignments. For each following programming assignment, you will receive a new `assignmentXX` archive to be unpacked into your `basic-techniques` folder. Note that the folder `assignment01` is already present.

```
basic-techniques
├── assignment01/
│   ├── assignment.cc
│   ├── assignment.hh
│   ├── main.cc
│   ├── MEMBERS.txt
│   └── ...
├── assignmentXX/
├── ...
├── common/
├── extern/
├── CMakeLists.txt
└── README.md
```

(b)   Compile

Our programming assignments require CMake ≥ 3.12 as well as a C++11 compiler to be installed. In the `basic-techniques` folder, type `mkdir build` and `cd build` to create and switch to a build folder. Then type `cmake ../` to generate makefiles for your platform.

On Linux, type `make` to compile the assignment. To run the assignment enter `./assignment01`

Note that the assignments are written with Linux as a target in mind. You can try using Windows or Mac at your own risk. (On Windows, you should find a Visual Studio file `basic-techniques.sln` in your build folder after running cmake.)

When running `./assignment01` you should see a black window, as well as one line of terminal output starting with `Loaded OpenGL 3.3 Context ...`

Though not required, we suggest that every student individually compiles and runs the application on the computer they intend to use for future programming assignments in order to get acquainted with our programming framework. (Yet, only one solution must be submitted per team.)

(c)   Add Debug Output

In the file `assignment.cc`, inside the function `drawScene(...)` print the current scene id and time to `std::cout`. Don't forget to insert new lines using `std::endl`.

(d)   Specify Group Members

In this course, we will use the Moodle group submission feature. In addition, you are required to enter the student ids and names of all group members in the file `MEMBERS.txt`. This step is necessary in every assignment. Remember, that only groups of 3-4 students are allowed. In case your group loses members, please regroup!

(e) Submit Your Solution

Follow the general instructions for programming assignments laid out on the first page and also explained in the tutorial course. This includes correctly submitting your solution through Moodle. Since you changed the files `assignment.cc` and `MEMBERS.txt` your solution zip archive will contain exactly two files: `assignment01/assignment.cc` and `assignment01/MEMBERS.txt`. Only one group member has to submit!

Note: The test program demonstrates that you can compile a simple OpenGL program with the C++ development environment of your choice. It also gives you information about your computer's graphics card and drivers.

For the rare case that your system does not meet the requirements to run our programming assignments (i.e. your GPU does not support OpenGL 3.3 or higher), we prepared a virtual machine for you as a fallback. Please only use the machine as a last resort, since it has very limited performance! (OpenGL will be emulated in software.) You can download the machine from the course website `https://www.graphics.rwth-aachen.de/course/193/`.