Prof. Dr. Leif Kobbelt
Kersten Schuster, Philip Trettner

Visual Computing Institute | RWTH AACHEN UNIVERSITY

Real-time Graphics Assignment 1

Date Published: October 25th 2017, Date Due: November 8th 2017, 11:44

- The assignments have to be done in groups of 2 students.
- Hand in the solutions to the exercises via L^{2}P.
- You are only allowed to change code inside the marked strips (STUDENT CODE BEGIN/END)!
- Any questions? → L^{2}P discussion forum or rtg@cs.rwth-aachen.de!

The **only** files that you should modify and **upload**:

- Assignment01.cc

Exercise 1 Compiling and Hand In [3 Points]

Obtain the (publicly accessible) exercise framework and assignments from
`https://www.graphics.rwth-aachen.de:9000/Teaching/rtg-ws17-assignments/`.
If you do not want to use git, you can also download the code as a zip file from there.
Once the code was cloned (or downloaded), you will find build instructions in the `Readme.md`.

Points are given for code handed in via L^{2}P that is compiling and attempts to solve the second exercise.

Exercise 2 Main Loop [1+6 Points]

(a) Complete the `Assignment01::getGroup()` function. This function must return your group, i.e. all names and immatriculation numbers.

(b) Based on the information from `getGroup()`, the assignment chooses a variant of the Main Loop for you to implement. Your task is displayed when running the program. Note that this depends on your immatriculation numbers, so you should definitely form a proper group before starting this subtask. Complete the function `Assignment01::mainLoop()`. For your convenience, an exemplary implementation of the simplest variant (variable timestep) is given. Depending on your group, one of the following versions should be implemented:

- A variable timestep loop (both rendering and update) but with a maximum FPS. If the loop body is too fast, sleep for some time to reduce the CPU burden.
- A loop with fixed timestep updating but variable timestep rendering.
- A fixed timestep loop (both rendering and update) that drops frames if simulation time is too much out of sync. (In reality, one would not put rendering and update together in a fixed timestep. This is only intended as an exercise.)