SWS3005: Real-Time Graphics Rendering (2023)

Assignment #2 (Group Work)

Release Date: 14 July 2023, Friday

Submission Deadline: 18 July 2023, Tuesday, 11:59 PM

TASKS

You are to complete [illegible]n to draw a **skybox** and perform **normal mapping**, **procedural bump mapping** and **[illegible] mapping**. The following image shows a sample view of the result that your program is expected to produce:

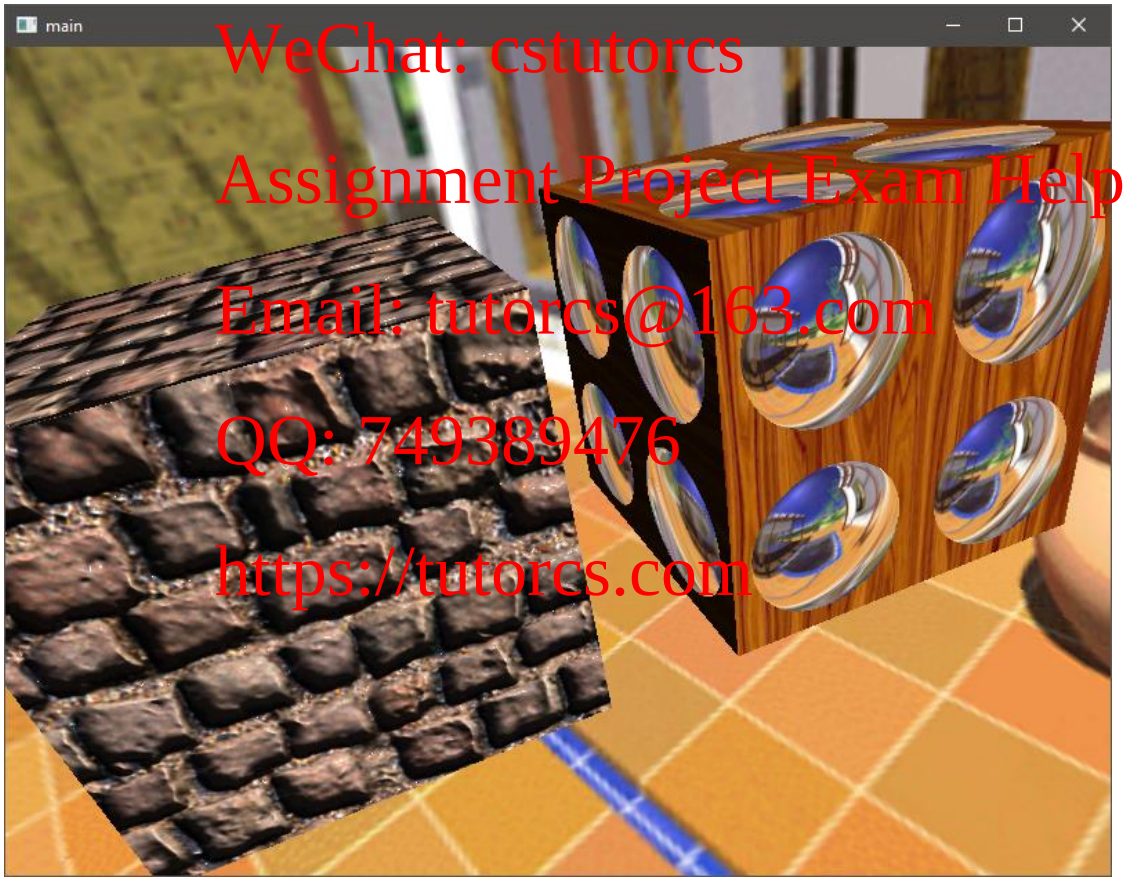

Figure 1

The background is part of a **skybox**. A skybox is used to show the supposedly-infinitely-faraway surroundings of a 3D scene. It is made of a 3D cube texture-mapped with an environment map (a cubemap), and the cube encloses the 3D scene and the camera.

The cube on the left side of the sample image is rendered with a brick texture map and a corresponding normal map. The color texture map is used to provide the ambient and diffuse material for the Phong lighting computation.

The cube on the right is rendered with a wood texture map, where it provides the ambient and diffuse material for the Phong lighting computation. The cube also has an array of hemispherical mirrors, and these mirrors reflect the environment (same environment map as used for the skybox). The

mirrors are produced using procedural bump mapping, and their reflection of the environment is produced using reflection mapping with a cubemap.

Please download th…**s3005_2023_assign2_todo_(*).zip** from the **Canvas > SWS3005 …nments** folder. You can try the completed executable program **main_done.…** or **main_done** (for macOS), found in the downloaded ZIP file. The program …rect rendering right now since it is using the incomplete shaders.

You need to comple…ation program **main.cpp** and the fragment shader **shader.frag**. In the fragment shader, all necessary **uniform variables**, and **global input/output variables** have already been declared, and **you must not add new ones**. You can add new functions in your shader. Note that you should adhere to the **variable naming convention** where the prefix "**ec**" is used to indicate that the entity is expressed in the eye space, the prefix "**wc**" to indicate world space, and the prefix "**tan**" to indicate tangent space.

A Visual Studio 2017 solution **main.sln** (or XCode project **main.xcodeproj** on macOS) is provided for you to build the executable program. The application program loads the shader source files **shader.vert** and **shader.frag**, and use them in the rendering. It also provides the values for the **vertex attributes** and **uniform** variables to the shaders. In this assignment, **you are not required and must not change any other C/C++ source files** besides **main.cpp**.

There are **three separate tasks** in this assignment:

Task 1: Skybox

For this task, you need to modify **main.cpp** such that the skybox cube is sized and positioned correctly. Please read the instruction provided in the given code to find out the details. The skybox cube is to be texture-mapped with an environment cubemap, and this cubemap has already been set up by the application program. Please note that the view of the skybox will change only when the camera rotates, and should not change at all when the camera translates. This is because the environment on the skybox is supposed to be infinitely far away.

Task 2: Brick Cube

For this task, you need to modify **shader.frag** to render the brick cube using normal mapping. Please read the instruction provided in the given code to find out the details.

Task 3: Wooden Cube

For this task, you need to modify **shader.frag** to render the cube with wood texture and produce the array of hemispherical mirrors using procedural bump mapping. Please read the instruction provided in the given code to find out the details. Note that the bumps must appear as if they are hemispherical. In addition, the reflection of the environment must match the orientation of the skybox.

GRADING

The maximum marks for this programming assignment is **100**, and it constitutes 20% of your total marks for the course. The marks are allocated as follows:

- **Task 1 — 20 marks.**
- **Task 2 — 40 m**
- **Task 3 — 40 m**

Note that marks will be ing style. If your program cannot be compiled and linked, you get 0 (zero) mark.

Good coding style. C adequately, use meaningful names for functions and variables (adhere to the new variable naming convention), and indent your code properly. You must fill in your **group number**, and every group member's **name** and **NUS User ID** in the **header comment**.

SUBMISSION

For this assignment, you need to **submit only**

- Your completed `main.cpp` that contains code for Task 1;
- Your completed `shader.frag` that contains code for Task 2 and Task 3.

You must put them in a ZIP file and name your ZIP file `group<group#>_A2.zip`. For example, if your group number is **06**, you should name your file `group06_A2.zip`.

Submit your ZIP file to `Canvas > SWS3005 > Assignments > Assignment #2`. **Only one group member should submit the file**. Before the submission deadline, you may upload your ZIP file as many times as you want. **We will take only your latest submission**.

DEADLINE

Late submissions will NOT be accepted. The submission page will automatically close at the deadline.

——— End of Document ———