**COMP371: COMPUTER GRAPHICS** **SUMMER 2023**

**ACADEMIC YEAR: 2023-2024**

# Quiz 2

**Quiz Date: August 02, 2023**

**Quiz Time: 36 hours from Noon**

**Marks: 50**

## RULES:

**When done, submit via Moodle:**

- Submit all project, dependency, and source files (cpp, h, vcproj, sln, *etc.*)
- Do not delay the submission until the last minute! Even partial submission must still be uploaded.
- The submissions are allowed to be re-uploaded multiple times. The latest uploaded will be used for marking within the timeframe.

During the quiz you are allowed to have any online or other media resources about OpenGL and code samples you can lay your hands on, *etc.* BUT:

- You are not allowed to share any quiz related material anytime. At best, the penalties for giving and taking each other's quiz code will result in 0 grade by default. Later penalties may follow.
- You are not allowed to communicate with each other at all orally, electronically, or otherwise during quiz time. You are not allowed to seek any help from anywhere.
- Should you use some code from external resources, URLs to those resources must be present within the code that you submit as comments in the header.
- Your own code from A1 as well as Q1 is allowed to be used as it is assumed to be well documented and referenced (if it is not so, this is another chance to fix that).

**Any form of Plagiarism will be strictly penalized.**

## TASKS:

**Comments in the code are of paramount importance.**

• Briefly (but identifiably) cite all the resources used (*e.g.*, online as URLs or offline as book or article titles, notes, own project *etc.*)

• Certain tasks need to be done in order, but some are not. Read through them all; do the sequences you absolutely need and the ones you are most comfortable with first, and then do as many as you can:

You will be building up on Quiz 1 and below mentioned requirements are already available to you through Quiz1 and A2 code.

Below mentioned requirements are in addition to the baseline code from Quiz 1.

- Create three cubical poles to hold the tennis net in place. Two on the two ends and one in the middle. Apply a texture to each of the poles as well as to the top row of the tennis net. You must make use of the repeat texture parameter. **(6 Pts)**
- Bring the four alphabets modeled in quiz 1 together on top of just two rackets (two to be placed on top of each racket). The front face of each alphabet must be transparent (Use a suitable alpha value). **(6 Pts)**
  1. The two models should be centered on either side of the tennis net.
  2. Each racket and corresponding two alphabets must be modeled as one entity.
  3. This will help you transform the racket and the alphabets together.
  4. You are encouraged to use your cube utility functions/methods/classes from A1, Q1.
  5. Apply a distinct texture to each alphabet, a metallic texture to the racket structure, and a tattooed texture to arm.
- Apply a suitable texture to the sky cube (large cube modeled in Q1) from inside. **(2 Pts)**
- Add a light source (make sure it is a spotlight, using the Phong illumination model) and place it 30 units out on the z-axis facing the model. Allow for it to be turned on and off with 'L'. Ambient light is necessary to be turned on when the main light source at the top is off. **(4 Pts)**
- Add one camera each onto each racket model's front: unobscured. Assign key 'M' to loop through the available cameras. Now you have 3 cameras (default main and the 2 with the models.). 'R' from Quiz 1 resets back to the main camera among other things. **(6 Pts)**
- Add another camera which will circle around the center of the scene using arrow keys. Attach a spotlight to this camera which will move with the camera and can be toggled using a key. Choice of the light constants, material properties, and color is left to you. Circling camera's Y and the look at point must remain constant. **(5 Pts)**
- Render the scene with shadows using two pass shadow algorithms. You don't have to consider the spotlight from the previous point for the shadow computation (Define a key to toggle it on and off). **(6 Pts)**
- The application must use OpenGL 4.1 and onwards and must include brief comments explaining each step. Grader will ask for a small modification to understand your grasp of your code. **(15 Pts)**

## Submission:

Quiz must be submitted through Moodle. No other form of submission will be considered. Please create a zip file containing your C/C++ code, vertex shader, fragment shader, a readme file (.txt). The zip file should be named Assignment#_YourStudetID. In the readme file document, the features and functionality of the application, and anything else you want the grader to know *i.e.,* control keys, keyboard/mouse shortcuts, *etc.*

## Bonus Feature(s):

You can achieve an extra 5 bonus points if you can create a randomized rotation pattern for the spotlight facing the model. This bonus will be given at Grader's discretion.

## Evaluation Procedure

You MUST demonstrate your program to the grader during a pre-scheduled session. You must run your code submitted before the deadline, demonstrate its full functionality, and answer questions about the OpenGL programming aspects of your solution. Major marking is done on the spot during the demo. Your code will be further checked for structure, non-plagiarism, *etc.* However, ONLY demonstrated submissions will receive marks. Other submissions will receive a zero score.