CS4247   Graphics Rendering Techniques   (2020/2021 Semester 2)

Assignment 3

Release Date: 19 March 2021, Friday
Submission Deadline: 4 April 2021, Sunday, 11:59 PM

TASK 1: Whitted Ray Tracing

You are to complete a Shadertoy GLSL fragment shader that implements the Whitted Ray Tracing algorithm. Your completed shader is to be run at https://www.shadertoy.com/new. The following images show sample views of the result that your program is expected to produce:
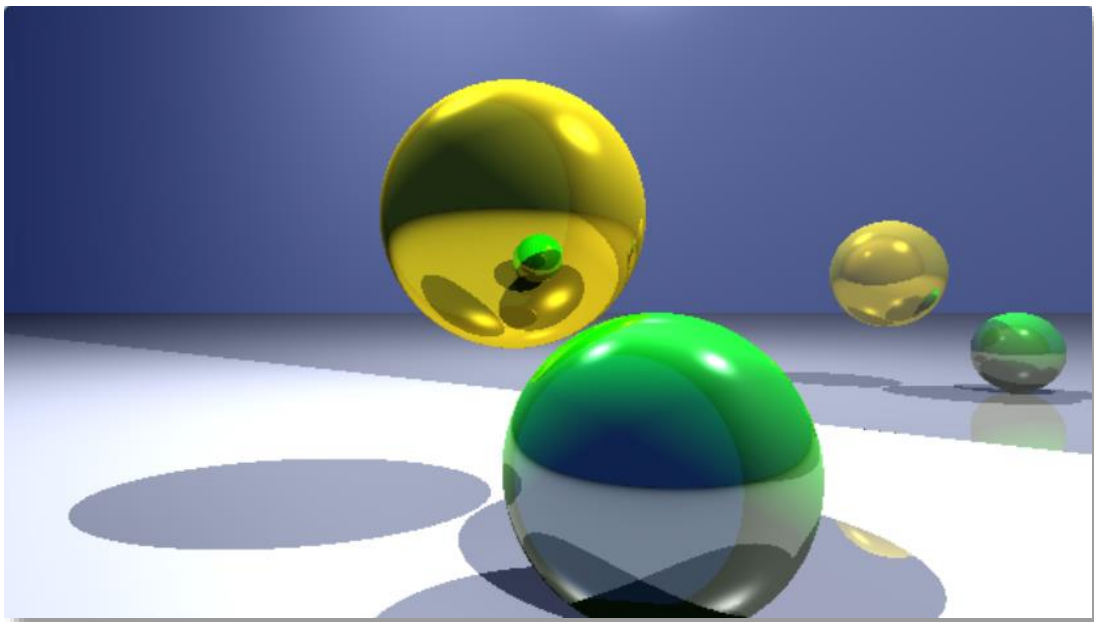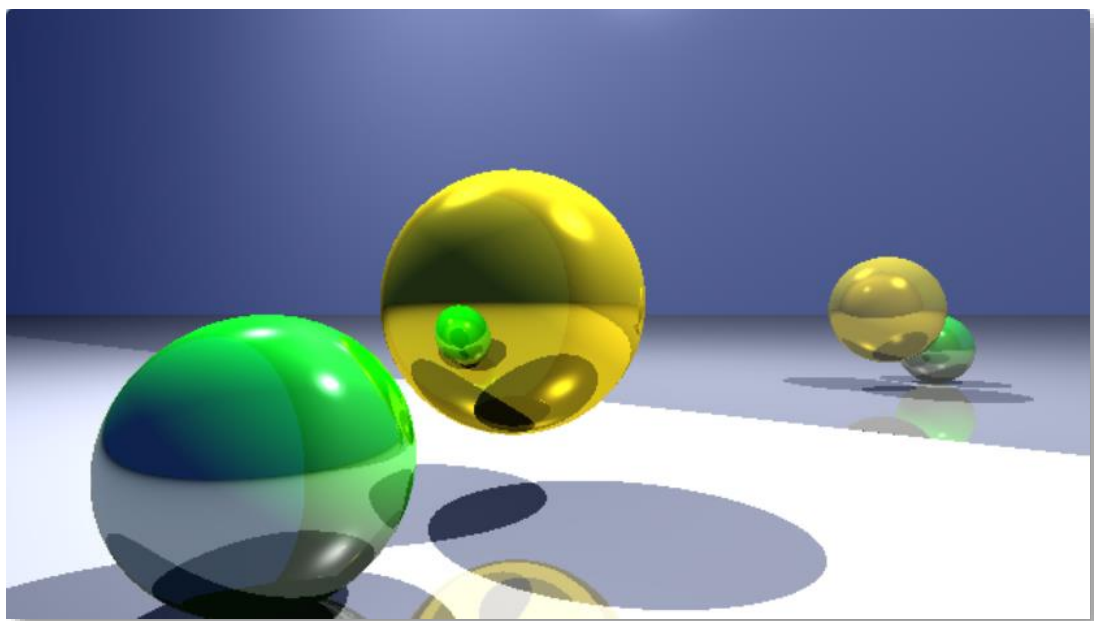

Figure 1

Figure 2

The scene consists of a silvery horizontal plane, a silvery vertical plane (in the background), a bouncing golden yellow ball, and a revolving shiny green ball. There are two point light sources. The images above were produced using 2 levels of ray tracing (recursion level = 2).

Please download the ZIP file **cs4247_2021S2_assign3_todo.zip** from the **Assignments** folder in LumiNUS Files.

You will find an incomplete Shadertoy GLSL fragment shader in the file **tasks_todo.frag**. You need to complete **only the parts relevant to Task 1** and save your completed shader in the file **task1.frag**. To run your shader, you need to copy-and-paste the shader source code to the source editing window at https://www.shadertoy.com/new. To run or rerun your shader, you just need to click the small black triangle at the bottom-left corner of the source editing window, and the result will be shown in the rendered image on the left of the page. You can do your coding and editing in the Shadertoy webpage, but make sure you copy the updated source code back to the file **task1.frag**.

The detailed instructions and requirements for completing the fragment shader can be found in the given source code. A brief introduction to how to write shaders for Shadertoy can be found at https://www.shadertoy.com/howto.

Run your completed shader in Shadertoy and screen-capture a frame of the rendered animation. The captured image must be at least of size 640x360. Save the image in PNG format as **task1.png**.

TASK 2: Distribution Ray Tracing

Based on your shader for Task 1, extend it by implementing the **Distribution Ray Tracing** algorithm to produce **depth-of-field effect** and **image anti-aliasing** (reduction of jaggies).

Note that it is not necessary to use stratified (jittered) sampling to generate random points on the lens and inside the pixel.

The detailed instructions and requirements for completing this task can be found in the given source code. Save your completed shader for Task 2 in the file **task2.frag**.

Run your completed shader in Shadertoy and screen-capture a frame of the rendered animation. The captured image must be at least of size 640x360. Save the image in PNG format as **task2.png**.

The following images show sample views of the result that your program is expected to produce:
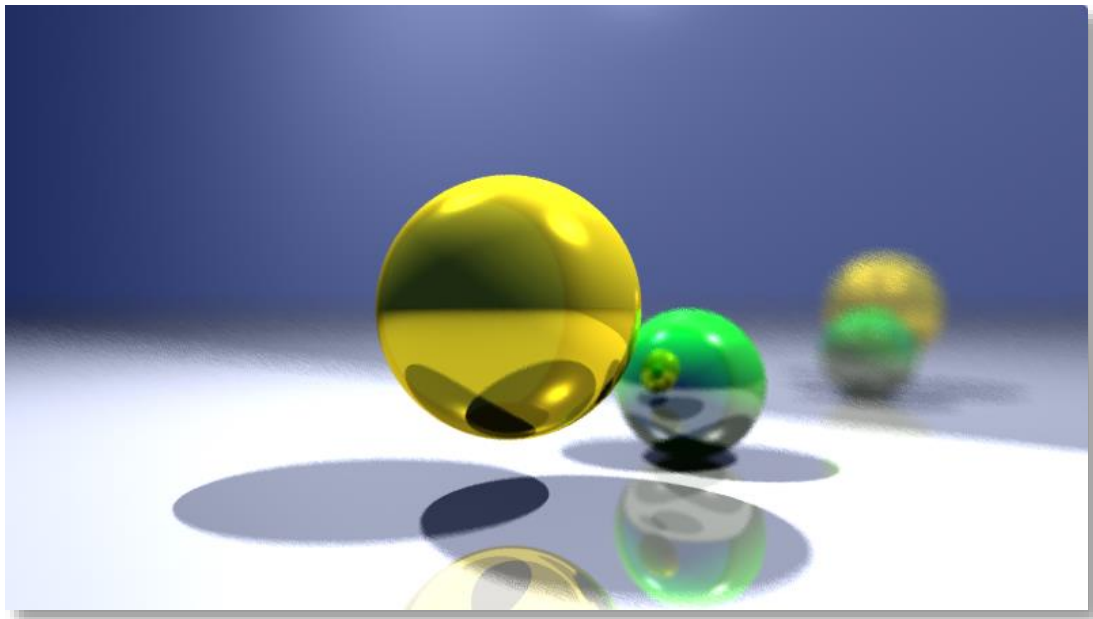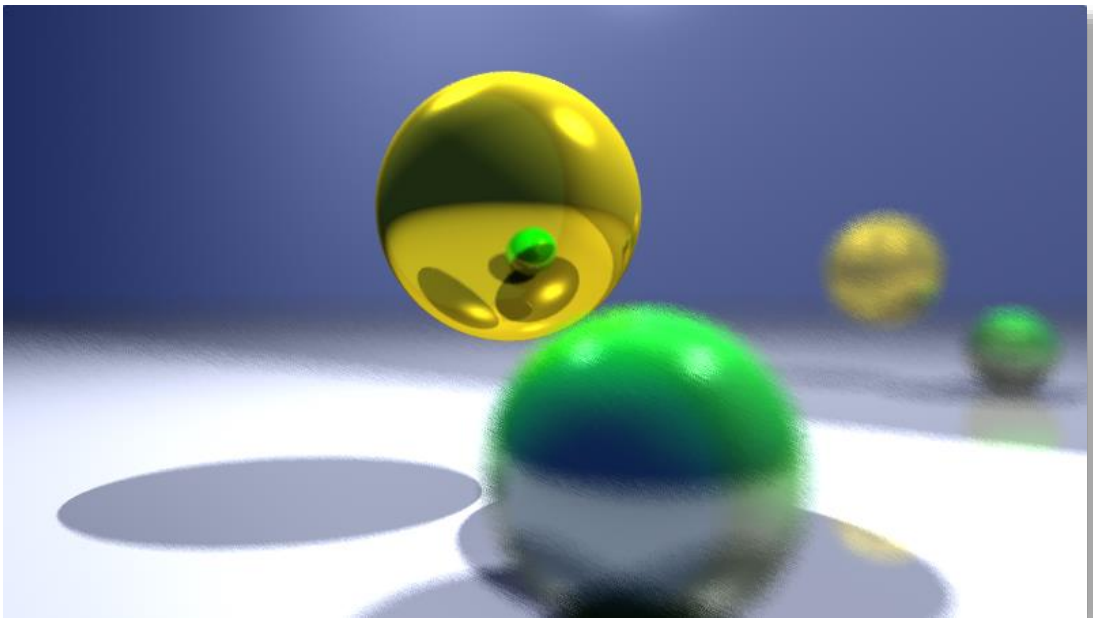
Figure 3

Figure 4

TASK 3: Design Your New Scene

For this task, you are to model and render a new 3D scene by modifying the shader you have completed for **Task 1 or Task 2**. Your new scene should not look similar to the original scene. Moreover, at least one of the objects, and at least one of the light sources **must be animated**. The animation must be perpetual.

Put your completed shader code in the file **task3.frag**. Your work will be assessed by the aesthetics and creativeness of the new scene and the animation.

Run your completed shader in Shadertoy and screen-capture two frames of the rendered animation. The captured images must be at least of size 640x360. Save the images in PNG format as **task3a.png** and **task3b.png**.

GRADING

The maximum marks for this programming assignment is **100**, and it constitutes **9%** of your total marks for the module. The marks are allocated as follows:

- **Task 1 — 45 marks,**
- **Task 2 — 40 marks,**
- **Task 3 — 15 marks.**

Note that marks will be deducted for bad coding style. If your shader cannot be compiled by Shadertoy, you get 0 (zero) mark.

Good coding style. Comment your code 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 **Name**, **Student Number**, and **NUS email address** in the **header comment**.

SUBMISSION

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

- **Task 1** — your completed **task1.frag** and captured frame **task1.png**;
- **Task 2** — your completed **task2.frag** and captured frame **task2.png**;
- **Task 3** — your completed **task3.frag** and captured frames **task3a.png** and **task3b.png**;

You must put it/them in a ZIP file and name your ZIP file *your-student-number_***assign3.zip**. For example, **A0123456X_assign3.zip**. All letters in your student number must be capitalized.

Submit your ZIP file to the **Assignment 3 Submission** folder in LumiNUS Files. Before the submission deadline, you may upload your ZIP file as many times as you want to the correct folder. **We will take only your latest submission.** If you have uploaded a new version to the folder, you **must delete the older versions**.

DEADLINE

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

See next page for Bonus Task.

BONUS TASK (OPTIONAL)

This bonus task is **completely optional**. You can still get full marks for the module without doing it.

The maximum marks for this bonus task is **100**, and it adds up to **2 additional marks to your total CA marks (maximum of 60 marks)** of CS4247. If, as a result of the bonus marks, your total CA marks exceed 60, then it will be capped at 60.

Requirements

For this bonus task, you are required to extend the Distribution Ray Tracing algorithm you did for Tasks 2 to produce **soft shadows** (due to area light sources) OR **blur glossy (specular) reflection**. Here are the additional requirements:

- For the rendering of **soft shadows**, you can model the area light sources as axis-aligned rectangles. The area light sources need not be rendered (i.e. they are invisible). There is no need to use stratified (jittered) sampling to sample positions on the light sources.

- For **blur glossy reflection**, you can just use any sensible range of directions for the randomly perturbed reflection rays. The size of this range of directions should be inversely related to the **shininess** of the surface material.

Save your shader in the file **task_bonus.frag**. Please indicate in the **header comment** what you have implemented.

Run your completed shader in Shadertoy and screen-capture two frames of the rendered animation. The captured images must be at least of size 640x360. Save the images in PNG format as **task_bonus_a.png** and **task_bonus_b.png**.

Submission & Deadline

You need to **submit only**

- Your completed **task_bonus.frag** and captured frames **task_bonus_a.png** and **task_bonus_b.png**.

You must put it/them in a ZIP file and name your ZIP file *your-student-number_bonus3.zip*. For example, **A0123456X_bonus3.zip**. All letters in your student number must be capitalized.

Submit your ZIP file to the **Assignment 3 Submission** folder in LumiNUS Files. Before the submission deadline, you may upload your ZIP file as many times as you want to the correct folder. **We will take only your latest submission.** If you have uploaded a new version to the folder, you **must delete the older versions**.

This bonus task has the **same submission deadline** as Tasks 1, 2 & 3.

——— **End of Document** ———