Assignment #4

CSD2170 Programming Massively Parallel Processors, Fall 2022

Due Date:	As specified on the moodle
Topics covered:	Shader programming, Tessellation Shader
Deliverables:	The submitted project files are the source code of your shader programs (`ellipsoid.tesc`, `ellipsoid.tese`, `ellipsoid.vert` and `ellipsoid.frag`) and C++ program (`main.cpp`). The files should be put in a folder and subsequently zipped according to the stipulations set out in the course syllabus.
Objectives:	Implementation of tessellation in shader program and Vulkan application. Learn how to write tessellation control shader and tessellation evaluation shader.

Objectives

This assignment is the implementation of tessellation shaders (TCS and TES) and Vulkan application for ellipsoid subdivision.

Implementation Requirement

In our lecture, we have discussed how to generate bezier curve using TCS and TES. We will generate an ellipsoid in tessellation shaders.

Geometry Model of Ellipsoid

An ellipsoid centered at the origin can be described by the following set of parametric equations:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{1}$$

We can turn x, y and z into spherical coordinates

$$\begin{aligned} x &= a\cos\phi\cos\theta \\ y &= b\sin\phi \\ z &= c\cos\phi\sin\theta \end{aligned} \tag{2}$$

where $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and $\theta \in [-\pi, \pi]$. The above is useful when you convert u and v coordinates into spherical coordinates.

You should not use Vulkan C++ to generate the control points other than the centre for ellipsoid[1]. Instead, tessellation shader should generate the control points for subdivision. Therefore the minimum information generated by `main.cpp` is the centre of the ellipsoid and its parameters a, b and c. You should allow the view of ellipsoid change when the camera moves.

[1]You may generate the patch control points for ellipsoid in main.cpp and pass to the tessellatin shader to subdivide further. In this assignment, we do not adopt this approach.

Template of Tessellation Application

You should use the given Assignment 3 Part 2 Template to develop Vulkan tessellation application. Only `main.cpp` is required to modify. You may refer to the code shown in the slides.

Rubrics

This assignment will be graded over 100 points. Here is the breakdown:

- If your code fails to compile, zero is awarded immediately.
- If you did not use tessellation shader to generate the control points, zero is awarded immediately.
- Comments/Code Readability. Comments are important for the code to enhance readability. Up to 10 points will be deducted for insufficient comments.
- Correctness. You must ensure that the code works for different size of ellipsoid.
- Latency. Frame rate will be tested for the given samples. 20 points are allocated for performance evaluation.