Assignment 2: Folding and unfolding a Chinese hand fan

Introduction: In this assignment, you will model and color a Chinese hand fan. Next, you will be able to fold/unfold the fan with keyboard interaction.

Compile and run the program: 2 pts

In order to compile and run the program, you need to have **GLM** library installed. OpenGL Mathematics (GLM) is a header only C++ mathematics library for graphics software based on the OpenGL Shading Language (GLSL) specifications. Don't panic! As **GLM** is a header-only library, installation is simple. Go to the following website and download **GLM**.

http://glm.g-truc.net/0.9.7/index.html

Next, unzip the download. Copy and paste the directory (the innermost **glm** directory that contains some hpp file) into the same include directory where you placed GL during OpenGL installation. In Visual Studio 2019, it should be somewhere as follows:

C:\Program Files (x86)\Microsoft Visual Studio\2019\Professional\VC\Tools\MSVC\14.24.28314\include

You are done! In linux, you should already have GLM installed if you have followed the uploaded OpenGL installation guideline for linux.
Now, download the attached zipped file **ChineseFan.zip**; compile and run the program.

Add color as an attribute: 4 pts

Once you run the program and switch to the wireframe mode by pressing 'w', you can see the fan is made of 10 vertices. The picture below demonstrates how vertices form triangles. For example, triangle 1 is made of the following vertices: v0, v1, and v2, triangle 2 has the following three vertices i.e., v0, v2, and v3, and so on.

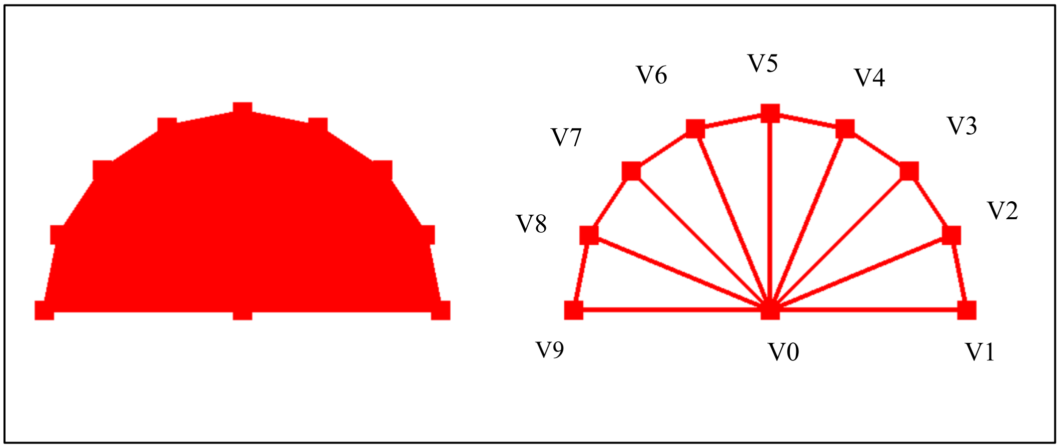
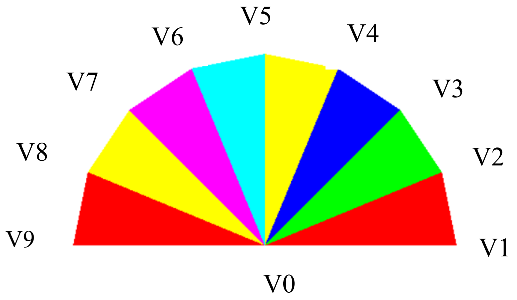


Figure 1: (Left) Surface-based and (right) wireframe-based representation of the fan.

Now, you need to add color to the fan so that different triangles are colored different as demonstrated in Figure 2. RGBA specifications for different colors are shown to the right of Figure 2.



Red (1.0, 0.0, 0.0, 1.0)
Green (0.0, 1.0, 0.0, 1.0)
Blue (0.0, 0.0, 1.0, 1.0)
Yellow (1.0, 1.0, 0.0, 1.0)
Cyan (0.0, 1.0, 1.0, 1.0)
Magenta(1.0, 0.0, 1.0, 1.0)

Figure 2: (a) Colored triangles in Chinese fan, and (b) RBGA specification of different colors.

Now, while creating the color attribute array, in order to be in conformance with Figure 2, you need to assign proper color values to different vertices. For example, the triangle composed of v0, v1, and v2 looks red, the next triangle composed of v0, v2, and v3 is green, and so on. For 8 triangles, you need to properly assign color to the vertices. **<u>You may find OpenGL tutorial 4 helpful.</u>** So, in the color array, you need to set values as follows:

```
GLfloat colors[] = { 1.0, 0.0, 0.0, 1.0, //0  - red
                     1.0, 0.0, 0.0, 1.0, //1  - red
                     1.0, 0.0, 0.0, 1.0, //2  - red

                     1.0, 0.0, 0.0, 1.0, //0  - green
                     1.0, 0.0, 0.0, 1.0, //2  - green
                     1.0, 0.0, 0.0, 1.0, //3 – green

                     1.0, 0.0, 0.0, 1.0, //0  - blue
                     1.0, 0.0, 0.0, 1.0, //3 -  blue
                     1.0, 0.0, 0.0, 1.0, //4 -  blue
                            :
                            :
                            :
                   };
```

Folding/unfolding the fan: **4 pts**

When the fan unfolds, the original vertex attribute array **vertices** is used. When folded, the vertex position attribute array '**vertices**' is replaced with '**vertices_updated**' array. You need to use **glBufferSubData** to switch between two arrays by toggling the keyboard button 'u'/'U'. **<u>You may find the uploaded UpdateSquare program helpful.</u>** Figure 3 demonstrates the unfolded and folded looks of the fan.

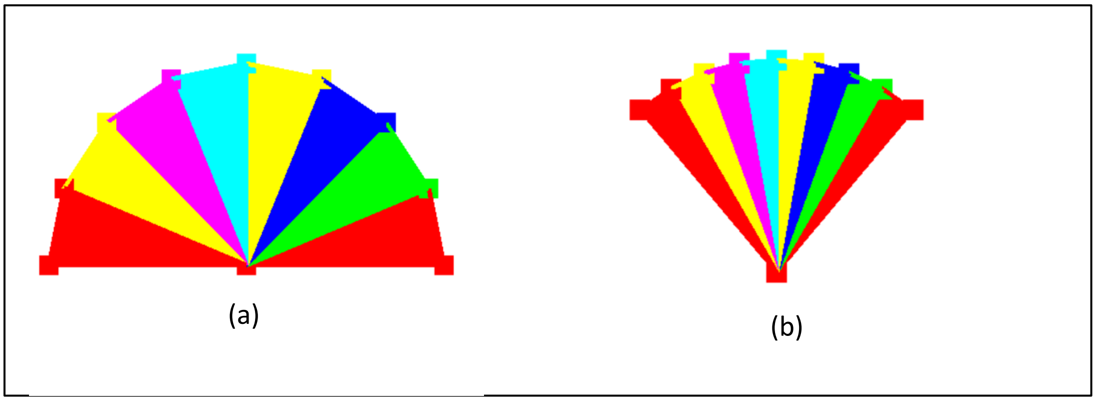


Figure 3: A Chinese hand fan (a) unfolded and (b) folded in the basic part.

Bonus part: **4 pts**

Use **glDrawElements** to draw the fan. You need to remove redundant vertices/ colors from the vertex and color attribute arrays. **You may find OpenGL tutorial 5 helpful.** Figure 4 shows the results. Please explain why each triangle does not exhibit distinct color when **glDrawElements** is used.

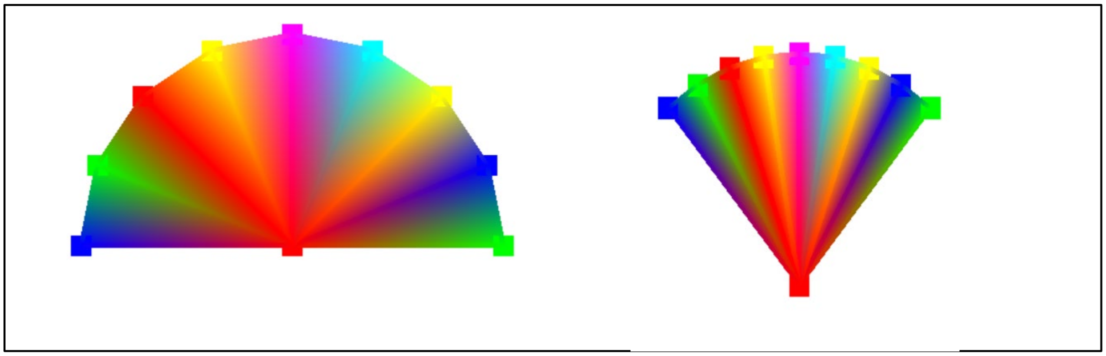


Figure 4: A Chinese hand fan (a) unfolded and (b) folded in the bonus part.

Submission:

Basic part: Compile and execute your code with g++ or the Microsoft compiler. Place your solution in a zipped file named with your last name followed by the first initial of your first name followed by 2 (ex: **YasminS2Basic.zip**) and submit the solution via canvas. The zipped file will contain the following files: ChineseFan.h, ChinesesFan.cpp, chineseFan.vs and chineseFan.fs.

Bonus part: In addition to the basic part, bonus part will be submitted in a zipped file named **BonusAssignment2.zip**. The zipped file will include the following files: ChineseFanBonus.h, ChinesFanBonus.cpp, chineseFan.vs and chineseFan.fs.

Submission deadline is **Wednesday, April 22, 11:59 pm**.

This assignment carries a weightage of **10%** of the course.