

Assignment 2: Instancing and Updating Polygons

In this assignment, you need to draw multiple copies of a hexagon using instancing. You need to use **glDrawElements**. Please find attached the file named **InstancedHexagon.cpp**. Please convert **glDrawArrays** to **glDrawElements**. You need to draw seven hexagons as specified in **hex_positions**. After implementation, you will be able to display the instanced hexagons as well as update the vertices of the hexagons with a keyboard button press, i.e., 'u'. For updating the vertices of the hexagons, you need to use the following array.

```
GLfloat hex_vertices_update[] = {  
    0.0, 0.0, 0.0, 1.0,  
    -0.1125, -0.1875, 0.0, 1.0,  
    0.1125, -0.1875, 0.0, 1.0,  
    0.1125, 0.0, 0.0, 1.0,  
    0.1125, 0.1875, 0.0, 1.0,  
    -0.1125, 0.1875, 0.0, 1.0,  
    -0.1125, 0.0, 0.0, 1.0  
};
```

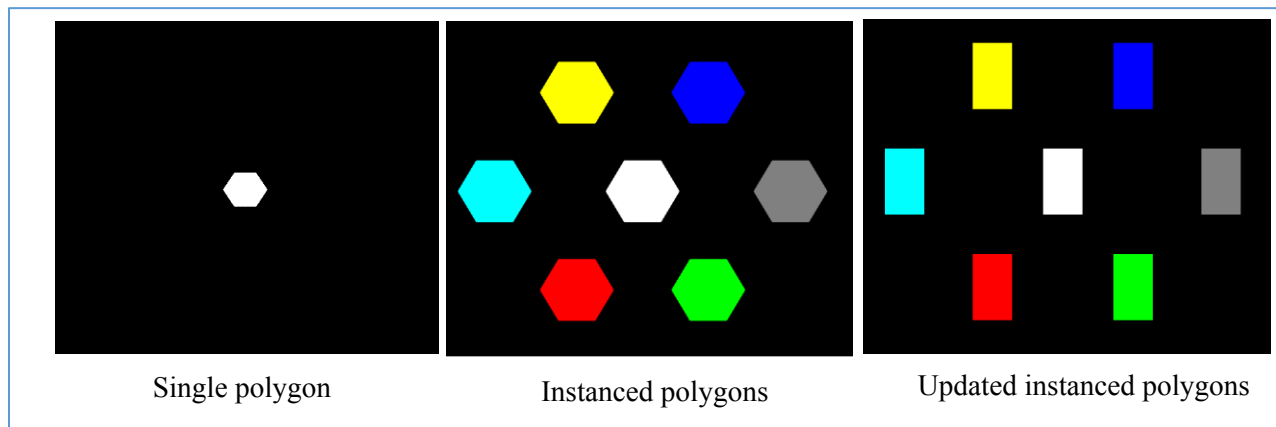


Figure 1: Instanced and updated polygons.

You'll also be able to draw the polygons in 'wireframe' mode with a keyboard button press 'f'. In wire frame mode, line width can be varied up to the maximum available line width. You will also be able to show the polygons with 'points only' options while varying the size of points upto the maximum point size with a keyboard button press 'p'.

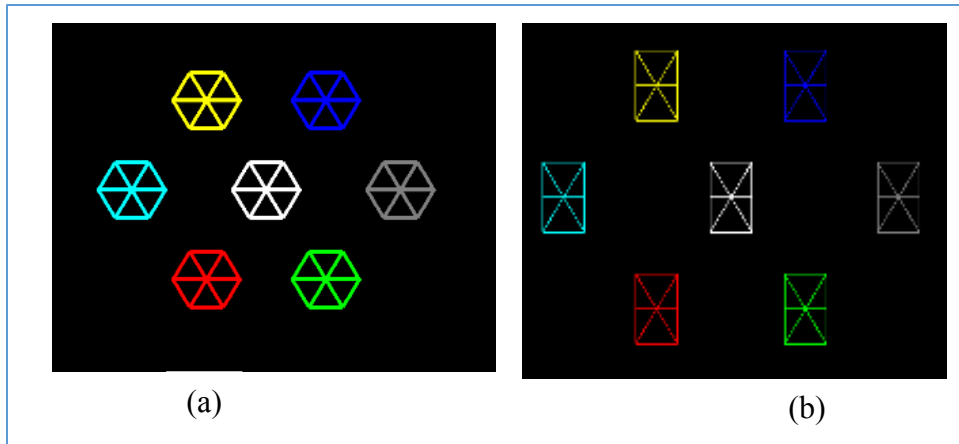


Figure 2: Polygons drawn in ‘wireframe’ mode with different line width.

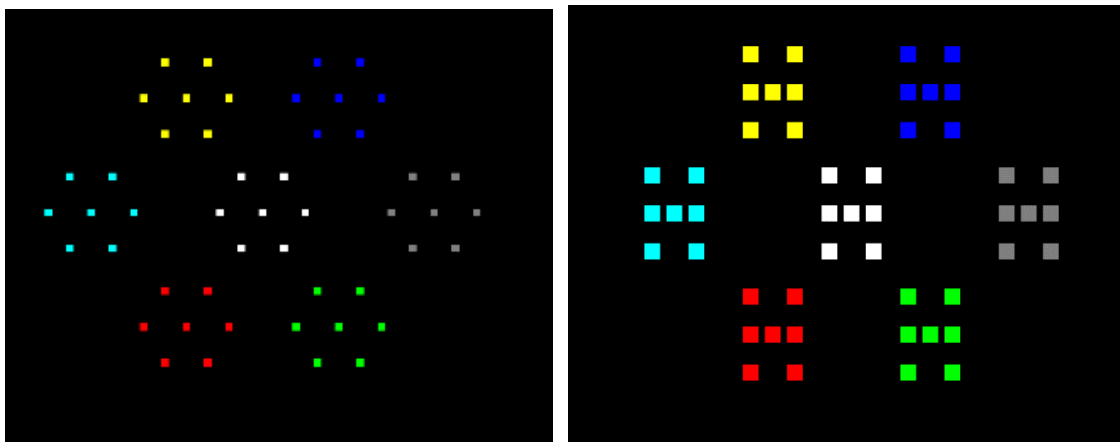


Figure 3: Polygons with different point sizes drawn with “Point only” options.

You’ll also be able to scale the polygon using a keyboard button press, i.e., ‘s’. You need to set some scale factor as follows:

vec3 scale(scale_x, scale_y, 1.0f); where **scale_x** and **scale_y** describe the scale values for x and y coordinate of the vertex. **scale_x** and **scale_y** can be randomly generated with each keyboard button press.

This scale factor will be multiplied with the vertices in vertex shader. You need to pass the scale factor as a uniform variable as follows:

glUniform3fv(scaleLoc, 1, &scale[0]);

where ‘**scaleLoc**’ is the location of the uniform variable.

Here are two examples of scaled polygons.

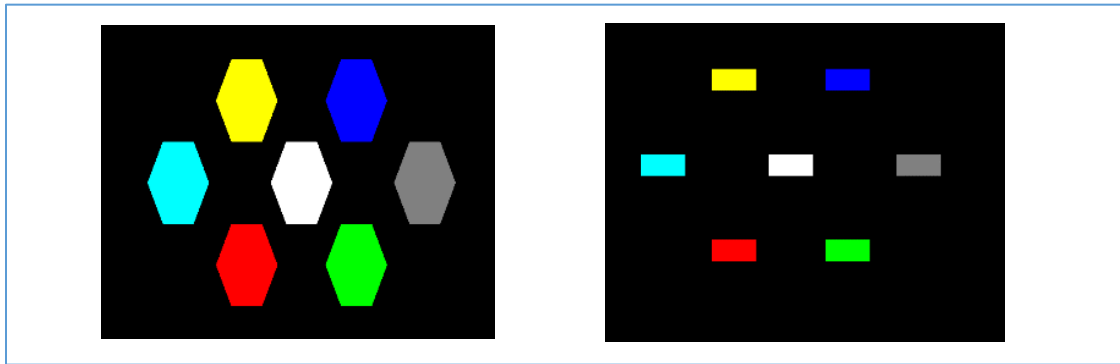


Figure 4: Scaled polygons.

Submission:

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: **YasminS2.zip**) and submit the solution via canvas.

Include at the top of your solution (in comments) your name, what compiler(s) you used. Minimal documentation is required – use good naming conventions for variables and comment any non- trivial code (describe what it does). Finally, capture your output from program. If necessary, you can include a text file that describes your approach towards implementing this program.

Thus, your zip should contain the following:

- **InstancedHexagon.cpp** and any additional file (if needed)
- output capture (name it something like **Instanced_hexagon_output**) and
- a text file describing your approach.

Submission deadline is **Friday, January 26, 11:59 pm**. Late submission will not be accepted.

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