Lab Assignment #4
Due Nov. 12, 2019

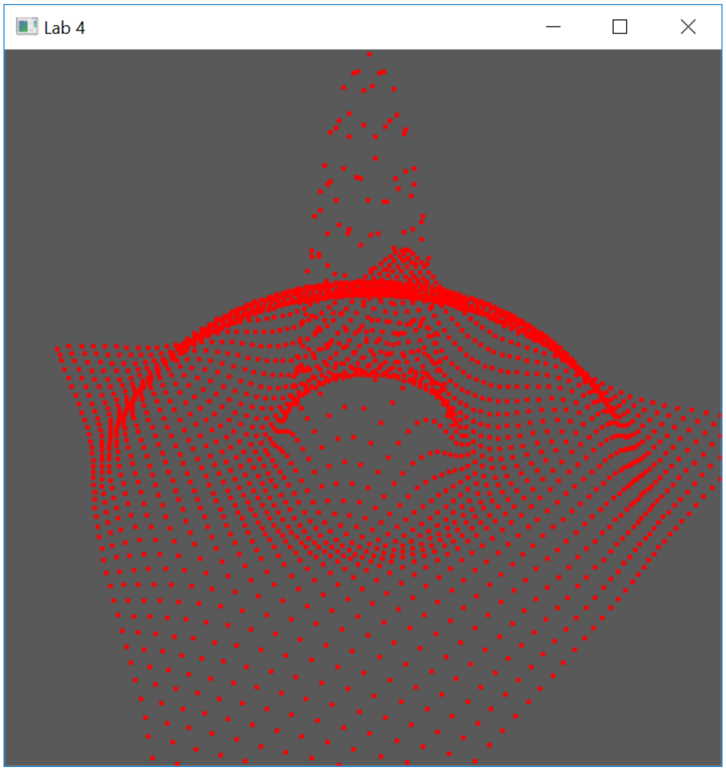
VBO and VAO

The goal of this lab is to get experience filling a vertex buffer object (VBO) with data, configuring a vertex array object (VAO), and drawing geometry using VAOs. Use the VAO Surf project as a starting point for this lab. You should be able to complete this assignment by modifying only the Surf.cpp file. The VAO Surf.cpp file calls the functions to create the VAO and draw its contents.

- Fill VBO. Modify the create_surf_vbo() function so that an NxN grid of points is inserted into the VBO. **[10 pts]**
    - The point locations should be determined using the surf(i,j) function. **[20 pts]**
    - When buffering data use the GL_STATIC_DRAW usage flag. **[10 pts]**

- Configure VAO. Modify the create_surf_vao() function. **[10 pts]**
    - In the glVertexAttribPointer function call, change the value of the stride parameter to 0. Although the stride between vertices is not actually 0, OpenGL interprets this value to mean that attributes are tightly packed into the buffer.

- Draw surface using VAO. Modify the draw_surf() function.
    - Bind the VAO and draw the buffer contents as GL_POINTS. **[10 pts]**
    - Make sure the count parameter is correct and the whole surface of NxN points is drawn. **[20 pts]** The surface should look like this:



- Draw the surface as triangles. Write new functions to create a VBO and VAO which hold the vertices of GL_TRIANGLES. **[5 pts]**
    - Create an imgui checkbox to toggle between this mode and drawing as GL_POINTS. **[5 pts]**
    - Change the OpenGL polygon mode (use glPolygonMode(…)) to GL_LINE when rendering in this mode so that the surface wireframe can be seen. **[10 pts]**

You will demo this program in class on the due date and submit your code to Blackboard. Be sure that you can demonstrate your program without recompiling code.