Ralenski Doucet Rendering Geometry pt2

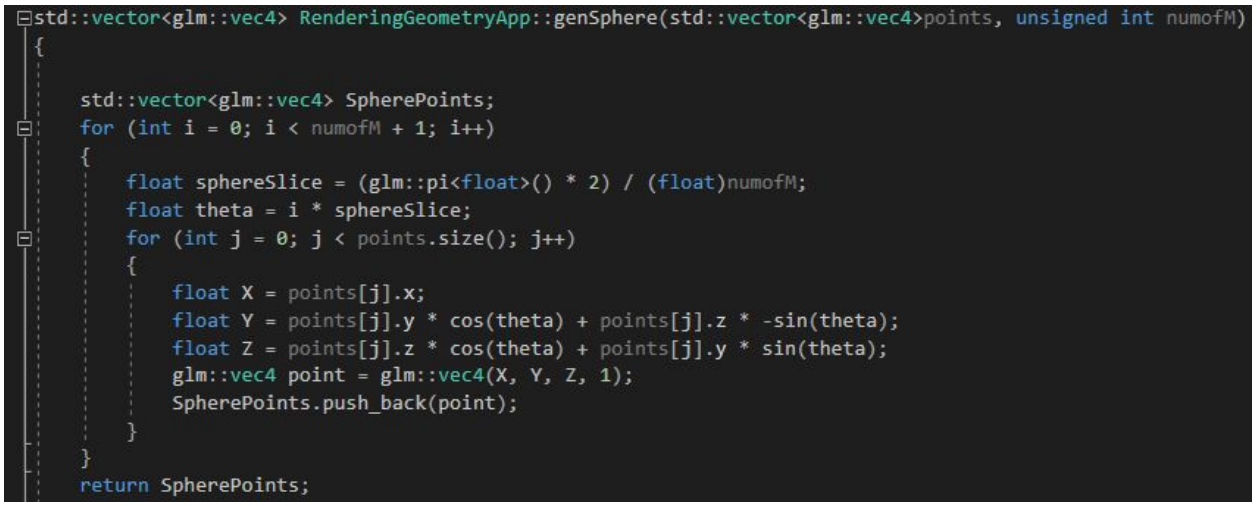
Problem::

2. Function that generates a sphere given a half circle, and number of meridians.

Answer::

To Generate a sphere you need to make function named gen sphere that is a std vector of type glm::vec 4 that takes in two arguments of std::vector of glm::vec4.

Then you create a local std::vector of type glm::vec4 i++.make a float named sphere splice that is assigned the value of glm::pi <float>()*2/float numofM.make a float named theta that equals the value of i*spheresplice and return speresplice.after you push back the sphere points.



```
std::vector<glm::vec4> RenderingGeometryApp::genSphere(std::vector<glm::vec4>points, unsigned int numofM)
{

    std::vector<glm::vec4> SpherePoints;
    for (int i = 0; i < numofM + 1; i++)
    {
        float sphereSlice = (glm::pi<float>() * 2) / (float)numofM;
        float theta = i * sphereSlice;
        for (int j = 0; j < points.size(); j++)
        {
            float X = points[j].x;
            float Y = points[j].y * cos(theta) + points[j].z * -sin(theta);
            float Z = points[j].z * cos(theta) + points[j].y * sin(theta);
            glm::vec4 point = glm::vec4(X, Y, Z, 1);
            SpherePoints.push_back(point);
        }
    }
    return SpherePoints;
```