Instructions

Epson makes a sensor module that can capture a golf swing and this test is about implementing the visualization of a golf swing in OpenGL **(See Final Output Demo).** The test has three parts:

1. Draw the golf swing data. The swing is recorded as coordinates in (x,y,z) and consists of two arc
    a. The first arc represents the wrist positions throughout the swing
    b. The second arc represents the golf club head positions throughout the swing

    The global variable **`g_MTracerData`** contains the swing data

```
static struct tMTracerData_
{
    // “data” is an array representing the points of the gofer's wrist
    (gripX, gripY and gripZ) and golf club head (headX, headY and HeadZ)
    struct Data {
        float gripX;
        float gripY;
        float gripZ;
        float headX;
        float headY;
        float headZ;
    } *data = nullptr;

    // Ignore this variable
    CMTracerFile  file;

    // Number of points for each arc. For example, if numPoints is 100,
    then there are 100 wrist points (100 gripX, gripY, gripZ) and 100 golf
    club head points (100 headX, headY and headZ)
    int           numPoints;

    // Ignore this variable
    mat4          transform;

    // Magic number that scales the golf club. Scaling the golf club with
    this factor will make it proportional to the golf swing
    float         clubScale;
} g_MTracerData;
```

2. Draw the golf club – Make it visible but no need to align it with the swing yet.

    The golf club is a triangle mesh implemented in club.h
    a. `Club::verts contains vertices of triangles (NOT triangular strip)`
    b. `Club::normal contains the normal to the triangles`
    c. `Club::numVert contains the number of triangles in verts`

3. Make the golf club move along the swing as shown in the demo solution (The data points are collected at 1000HZ). Note that, the rate displayed in the demo solution is incorrect.

Additional Information

1. Feel free to use Google to look for information

2. Implement the function RenderGL within render.cpp. The code to setup the rendering and mouse movement are already done, the test can be completed by implementing the following block within RenderGL in render.cpp

```
if (g_MTracerData.data && g_MTracerData.file.isValidFile())
{ … }
```

3. Club.h and club.cpp contains the triangle to the golf club

4. The solution demo is implemented with OpenGL 1.1