CPSC 4500 HW2

Due by end of day 21 January 2014

Your task

Write a program using OpenGL that implements an analog clock.

- It must have moving seconds, minutes, hours hands

- The face must be round

- Something must indicate hour positions

- Use multiple colors.

- Use different line widths and point sizes as appropriate.

- **Do not** use `GL_POLYGON`

Tips

Start with the provided `square.cpp`

Trigger a function call periodically

This is a technique used for animations. It works in Mac/Linux and **should** work in Windows. If it is determined to not work, I will find an alternate way to get the current hour,minute, second. Hardcode those values for now, and skip to writing `drawScene`.

This code defines a new subroutine `update` that updates global variables `hour`, `minute` and `second` every 100 ms. It finishes by telling Glut to do two things:

1. Redraw the scene

2. Call `update` again in 100 ms

```
#include <time.h>

static int hour=0;
static int minute=0;
static int second=0;

void update(int ignored)
{
    time_t now;
    struct tm* timeinfo;

    time(&now);
    timeinfo = localtime(&now);

    hour = timeinfo->tm_hour;
    minute = timeinfo->tm_min;
    second = timeinfo->tm_sec;

    glutPostRedisplay();         // tell glut to redraw the window
    glutTimerFunc(100, update, 0); // tell glut to call update again in 100 ms
}
```

To start this process, make sure you set up `update` in your main:

```
glutTimerFunc(100, update, 0);
glutMainLoop();
```

Write drawScene

Using the global variables `hour`, `minute` and `second`, your `drawScene` function should have everything it needs to draw the clock.

Turn in

Submit only your `.cpp` file – no Visual Studio projects! – to Blackboard, by the due date.