24780 Engineering Computation: Problem Set 5

(*) In the following instructions (and in all course materials), substitute your Andrew ID wherever you see *yourAndrewId*.

You need to create a ZIP file (which may appear as a compressed folder in Windows) and submit the ZIP file via the 24-780 Canvas. The filename of the ZIP file must be:

    PS05-YourAndrewID.zip

For example, if your Andrew account is hummingbird@andrew.cmu.edu, the filename must be:

    PS05-hummingbird.zip

Failure to comply with this naming rule will result in an automatic 5% deduction from this assignment's credit. If we cannot identify the submitter of the file, an additional 5% credit will be lost. If we are ultimately unable to connect you with the submitted ZIP file, you will receive 0 points for this assignment. Therefore, ensure strict adherence to this naming rule before submitting a file.

The ZIP file must be submitted to the 24-780 Canvas. If you find a mistake in a previous submission, you can re-submit the ZIP file with no penalty as long as it's before the submission deadline.

Your Zip file should contain only two files, ps5.cpp and ps5.pdf. Do not include project files and intermediate files generated by the compiler. But, do not worry about some files or directories that are automatically added by the archiver (__MACOSX__ file for example).

Notice: The grade will be assigned to the final submission only. In the case of multiple file submissions, earlier versions will be discarded. Therefore, when resubmitting a ZIP file, it MUST include all the required files. Also, if your final version is submitted after the submission deadline, the late-submission policy will be applied, regardless of how early your earlier version was submitted.

Ensure that your program can be compiled without errors on one of the compiler servers. Do not wait until the last minute, as the compiler servers may become very busy just minutes before the submission deadline!

Submission Due: Please refer to Canvas.

START EARLY!

Unless you are a good programmer, there is no way to finish the assignment overnight.

PS5 Go creative! [ps5.cpp] (100 pts)

Write a program that:

- uses OpenGL animation using double-buffering. (20 pts)

- runs the animation in a for or while loop until the user wants to terminate (for example, pressing ESC key. Or if you write a game program, it is also ok to run the program until the game is over.) (10 pts)

- must be an interactive program. (10 pts)

- uses at least one of the three features of OpenGL explained in class, color gradation, line stipple, or alpha blending. (20 pts)

- uses at least two types of OpenGL primitives from GL_POINTS, GL_LINES, GL_LINE_STRIP, GL_LINE_LOOP, GL_TRIANGLES, GL_TRIANGLE_STRIP, GL_TRIANGLE_FAN, GL_QUADS, GL_QUAD_STRIP, and GL_POLYGON. (20 pts)

- uses at least one of the following. (20 pts)

  - math library function,
  - shuffling,
  - sorting,
  - state transition,
  - numerical integration using Euler's method.

And submit:

- C++ source code (ps5.cpp)

- Brief description of your program that includes the following information in a PDF or plain-text.

  - How to use your program, and
  - Which features you used for satisfying the above requirements.
  - Include the document in the zip.

Our TAs and graders will select top 4 creative programs that will be shown in class.

Test Your Program with One of the Compiler Servers

Test your program with one of the following compiler servers:

```
http://freefood1.lan.local.cmu.edu
http://freefood2.lan.local.cmu.edu
http://freefood3.lan.local.cmu.edu
http://freefood4.lan.local.cmu.edu
```

You need to make sure you are not getting any errors (red lines) from the compiler server.

It is a good practice to remove warnings as well. However, we will not take points off for warnings as long as your program satisfies requirements of the assignment.

You can only access these servers from CMU network. If you need to access from your home, use CMU VPN. Please visit the CMU computing services web site how to install the VPN.