

Computer Graphics Assignment – Scene1 (S1)

Due on, September 18, 2020.

Submission Instructions:

1. Please submit your work directly on TRACS by the due date/time. Use only zip for compression.
2. Please write your name in the assignment header and as a part of the file name of files submitted.
3. It must be your own work – a penalty of at least one grade in your final grade and a report to the Dean of Students will result from sharing work or using other people work.
4. Please submit your assignment by the deadline – late submission will not be accepted and will result in a grade of 0. A grade of 1 denotes an issue with your assignments, which you have to resolve with the instructor.
5. **Please do not submit your assignment via email. If you have a justified documented reason for being late then please submit the assignment to your TRACS drop-box and notify me by email.**
6. Please submit only the source code of your program in C (C++) + OpenGL library functions. In addition, please submit an image (in JPEG format) that shows the scene obtained through running your program. If I have doubts or concerns about your program, I may request that you submit the entire files required to produce a working program under Linux so that I can test your submission.
7. The code should include remarks that explain any non-trivial part of the code. For example, if you use an 'n-gone' with 30 edges to generate a circle than you should explain it somewhere in the code. On the other hand, do not include explanations for basic GL functions such as glVertex etc.
8. Instructions on how to produce an image of your scene are given in assignment S0.

The goal of assignments S1 to S6 is to produce a scene of "medium complexity." For ideas about the default scene see color plate 24 in the book by Angel (available on TRACS).

The goal of this assignment (S1) is to produce a few basic objects. Consecutive assignments will manipulate these objects (e.g., duplicate, apply transformations, etc.) and the surroundings (light, shading) to create the actual scene.

Assignment Instructions:

- 1) Use only 3D parallel projection (glOrtho)
- 2) Define a "viewing volume" which is a cube centered at the origin with edge length of **3**.
- 3) Define a window of 800x600
- 4) Generate a Point, Line, Triangle, Square, Hexagon, Circle, and a cUbe, and interactively place them in the center of the viewing volume.
 - a. The objects should be normalized so that edges / radiuses are with length of 1. The length of an edge of the Hexagon should be 0.5.
 - b. The program interacts with the user. When the user hits T on the keyboard, the program cleans the screen, and places the triangle in the center of the viewing volume. Similarly, the program renders and places the rest of the objects according to keyboard strikes of "R, H, C, and U."
- 5) The Circle should be drawn using a 30-gon
- 6) There is no restriction on how to place the cube but notice that if you place it in a way that one of the faces are parallel to the Z=0 plan, then it will appear as a rectangular on the screen. A better effect would be if no face is parallel to the X=0, Y=0, and Z=0 plans.