

Computer Graphics Spring 2019

Assignment – Scene1 (S1)

Due on, February 17, 2019.

Submission Instructions:

1. Please submit your work directly in TRACS (using the TRACS editor) or as a text/MS-word/PDF attachment by the due date/time. Please use only zip for compression.
2. 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.
3. Please submit only the source code of your program in C (C++) + OpenGL / QT 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.
4. 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.
5. Please write your name in the assignment header and as a part of the file name of the attachment.
6. Please do not submit your assignment via email. If you miss the deadline, then please submit it on TRACS and send me an email notification.
7. A penalty of 10% per day will be assessed for late submission. In addition, it will be graded only at the end of the semester.
8. Instructions on how to produce an image of your scene are given below.

Assignment Instructions:

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. Following assignments will manipulate these objects (e.g., duplicate, apply transformations, etc) and the surroundings (light, shading) to create the actual scene.

- 1) Use only 3D parallel projection (glOrtho)
- 2) Define a "viewing volume" which is a cube centered at the origin with edge length of 4.
- 3) Define a window of 640x480
- 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 and centered so that edges / radius are with length of 1. The length of an edge of the Hexagon should be 0.5.
 - b. The program interacts with the user. For example, 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 "P, L, T, S, H, C, and U (U for Cube)."
 - c. The current best way, available for you, to approximate a circle is by drawing an n -gone with $n \geq 20$. To elaborate, an hexagon is an n -gone with $n = 6$, and an octagon is an n -gone with $n = 8$.
- 5) There is no restriction on how to place the cube, but notice that if you place it in a way that one (hence two) 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.

Producing an Image of your scene

The following are instructions for “screen dump”. That is, how to capture the screen / current window into a file under Windows and under Linux.

Print screen under Windows

(From: <http://www.entity.cc/ICONS/print-screen.php>)

To print the contents of the Screen, you must save an image of what's displayed on-screen to the clipboard and then paste it into a document where it can be printed.

(Windows Help: Print Screen)

To do this, follow this procedure:

1. Maximize the window you'd like to capture.
2. To copy/capture the current window, hold down **ALT + Print Scrn** at the same time. To capture the entire screen, just press **Print Scrn**.
3. Open a new document in MS Word, MS Photo Editor, or Adobe PhotoShop.
4. **Paste** the screen shot by holding down **CTRL** and **V** at the same time.
5. Print the document when you're finished pasting screen shots.

Print screen under Linux

(From: <http://www.sb.fsu.edu/~xray/Manuals/ScreenCapture.html>)

The same location includes more and allegedly better utilities for screen capture)

Using Linux's native utility 'xwd/xwud'

Most Linux and UNIX operating systems have native utilities called xwd (x-window dump) and xwud (x-window un-dump) that allow the user to capture either the whole screen or a specific window. In order for the user to capture specific window, issue the following command,

'xwd >myimage.xwd'.

Then click with the mouse inside the window of choice (if the whole screen needs to be captured, simply click anywhere in the screen). The utility will write out an image in the special '.xwd' format. The created image can be viewed either using the related utility 'xwud' or using any number of image processing software utilities such as, display, gimp, or xv. To view using 'xwud' issue the following command,

'xwud -in myimage.xwd<u1:p></u1:'

The image can be converted to .jpg or .png using the convert utility.

'convert myimage.xwd myimage.jpg'