

Module 1: Introduction and Graphics Primitives

Lab Exercises for Week 3

(a) Sierpinski Gasket

- Create a window that is one quarter the size of the screen, and position it in the middle of the screen.
- Set the background colour, and set your points to be drawn with size 3.0.
- Start with a triangle, with vertices v_0, v_1, v_2 being $\{-0.8, -0.8\}$, $\{0.0, 0.8\}$, and $\{0.8, -0.8\}$, and use $v_1 = \{0.0, 0.8\}$ as your starting point. The outline of the triangle should be drawn in Red with a line width of 5.0.
- Draw 1000 points in the Sierpinski gasket. The Sierpinski algorithm is
 - form a triangle using 3 fixed points (vertices v_0, v_1, v_2)
 - choose one of the vertices v_0, v_1, v_2 as a starting point (say v_1), and call it p_0
 - now iterate until the pattern is displayed “well-enough”:
 - choose one of the vertices v_0, v_1, v_2 at random, and call it v_R
 - form the new point p_i to be the midpoint between v_R and p_{i-1}
 - draw the point p_i
- Generate a new set of 1000 points whenever the SPACE key is pressed.
- Exit the program when ESC is pressed (`#define KEY_ESC 27`).
- Include special keys routine F1 (`GLUT_KEY_F1`), which will display ProgramHelp (namely, the functionality of the keyboard input SPACE and ESCAPE).
- Include a checkError routine to scan for errors during the display process, to be called at the end of your “display” function.

(b) Drawing a shape

- Draw a shape based on a hexagon with triangles on the hexagon edges, to resemble the figure below, using various Graphics Primitives.
- The vertices of the hexagon are: $(1,0)$, $(0.5,1)$, $(-0.5,1)$, $(-1,0)$, $(-0.5,-1)$, $(0.5,-1)$.
- Use simple geometry to determine the triangle vertices. The triangles are to be equilateral triangles.
- Assign the colours as indicated.
- Toggle between flat and smooth shading when pressing the space bar.
- Your programs from now on should always include ProgramHelp (via F1) plus a checkError routine.

