



COMP5514

Computer Graphics in C/C++

Lab 04

Prof. George Baci

csgeorge@comp.polyu.edu.hk

www.comp.polyu.edu.hk/~csgeorge/comp5514/lab/

Department of Computing --- The
Hong Kong Polytechnic University

Contents: OpenGL Points and Lines

PART A: OPENGL PRIMITIVES

- Points
- Lines
- Attributes

PART B: OPENGL FUNCTIONS

- GL Library
- GLU Library
- GLUT Library

PART C: EXAMPLES

- Basic OpenGL program

Recall: OpenGL/GLUT Program

◆ Recall your graphics program in “main.c”:

```
#include ... /* all the names of the header include files */
```

```
#define ... /* all the macro definitions */
```

```
int variable1 ... /* all the global types and variable declarations */
```

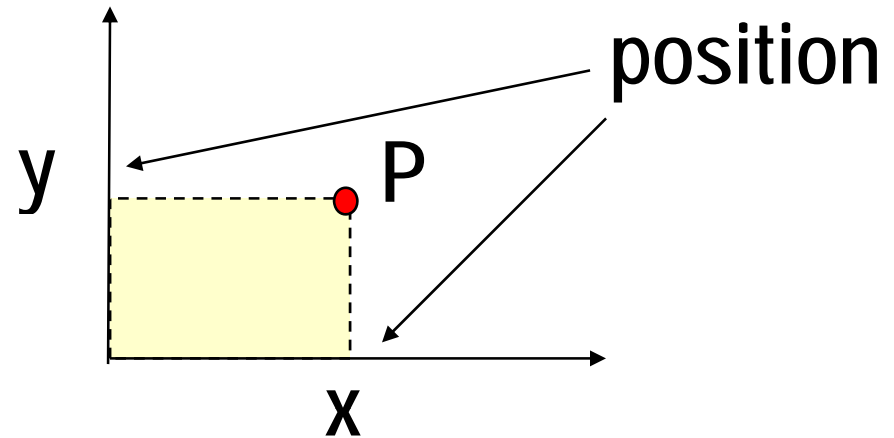
```
type functionA (...) { ... } /* all the function definitions */
```

```
void myDisplay (void) /* your graphics frame computation + opengl */  
{  
    /* OpenGL calls here */  
}
```

```
int main(int argc, char** argv) /* your main function */  
{  
    glutInit(argc, argv); /* graphics initialization */  
                          /* ... open window... */  
                          /* ... registration functions... */  
    glutDisplayFunc(myDisplay)  
    glutMainLoop();  
}
```

Geometric Primitives

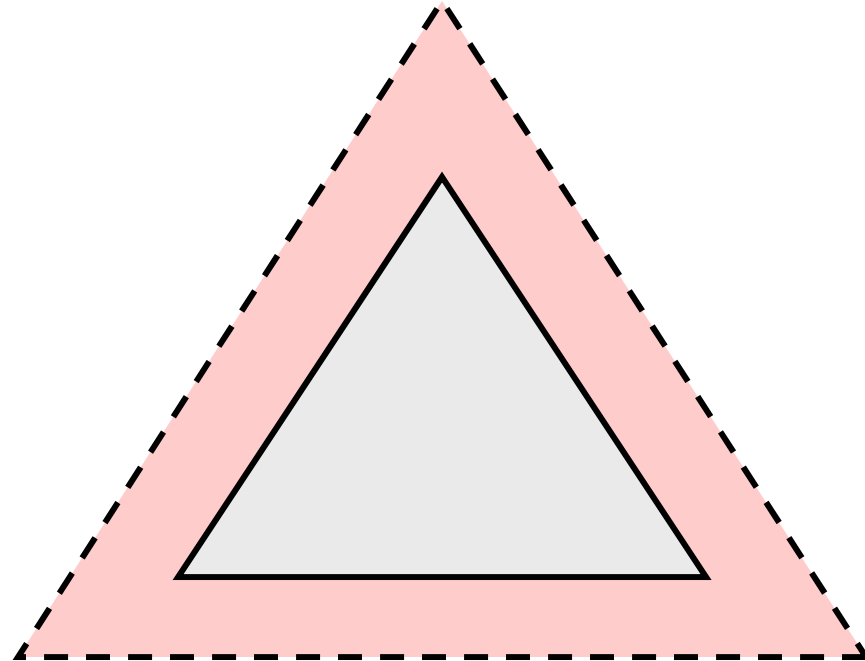
◆ Points



◆ Lines



◆ Triangles



Generating Points

```
glPushMatrix();           // Save the transformation matrix on stack
glRotatef(xRot, 1.0f, 0.0f, 0.0f);
glRotatef(yRot, 0.0f, 1.0f, 0.0f);
glBegin(GL_POINTS); // Begin points
z = -50.0f;               // Set the z-coordinate
for(angle = 0.0f; angle <= (2.0f*GL_PI)*3.0f; angle += 0.1f)
{
    x = 50.0f*sin(angle); // Change x
    y = 50.0f*cos(angle); // Change y
    glVertex3f(x, y, z);   // Specify the point
    z += 0.5f;            // Move the Z value up a little
}
glEnd();                 // End point list specifications
glPopMatrix();           // Restore transformations
glutSwapBuffers();        // Swap buffers
```

Generating Lines

```
glPushMatrix();           // Save the transformation matrix on stack
glRotatef(xRot, 1.0f, 0.0f, 0.0f);
glRotatef(yRot, 0.0f, 1.0f, 0.0f);
glBegin(GL_LINE_STRIP);    // Begin points
z = -50.0f;               // Set the z-coordinate
for(angle = 0.0f; angle <= (2.0f*GL_PI)*3.0f; angle += 0.1f)
{
    x = 50.0f*sin(angle); // Change x
    y = 50.0f*cos(angle); // Change y
    glVertex3f(x, y, z);   // Specify the point
    z += 0.5f;             // Move the Z value up a little
}
glEnd();                  // End point list specifications
glPopMatrix();            // Restore transformations
glutSwapBuffers();        // Swap buffers
```

Point Attributes

- ◆ Getting the current supported point size and granularity:

```
glGetFloatv(GL_POINT_SIZE_RANGE, sizes );
```

```
glGetFloatv(GL_POINT_SIZE_GRANULARITY, &step);
```

- ◆ In order to specify the current point size:

```
glPointSize(curSize);
```

References

- ◆ **Man pages:**

http://www.eecs.tulane.edu/www/graphics/doc/OpenGL-Man-Pages/ /opengl_index_alpha.html

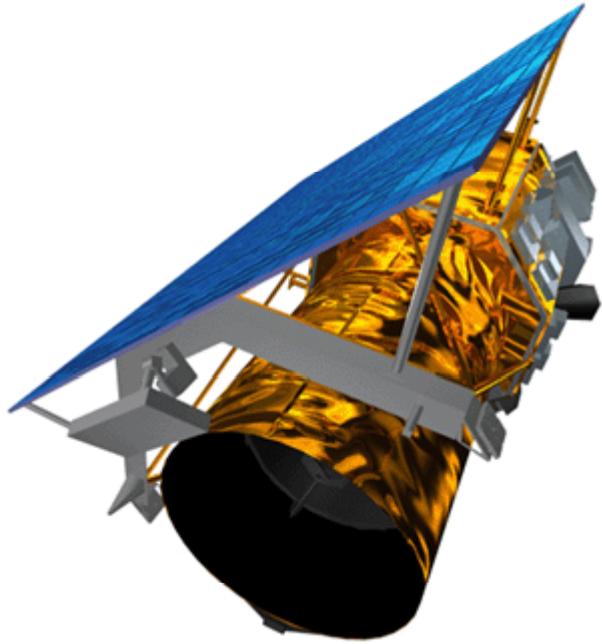
- ◆ **More on the web...**

Try things out...

- **Example:** `lab04.zip`



More on Images Later



The End

