Intoduction to OpenGL

- **What is OpenGL ?**

  - It is a standardized graphics library that has its roots in the Silicon Graphics GL library

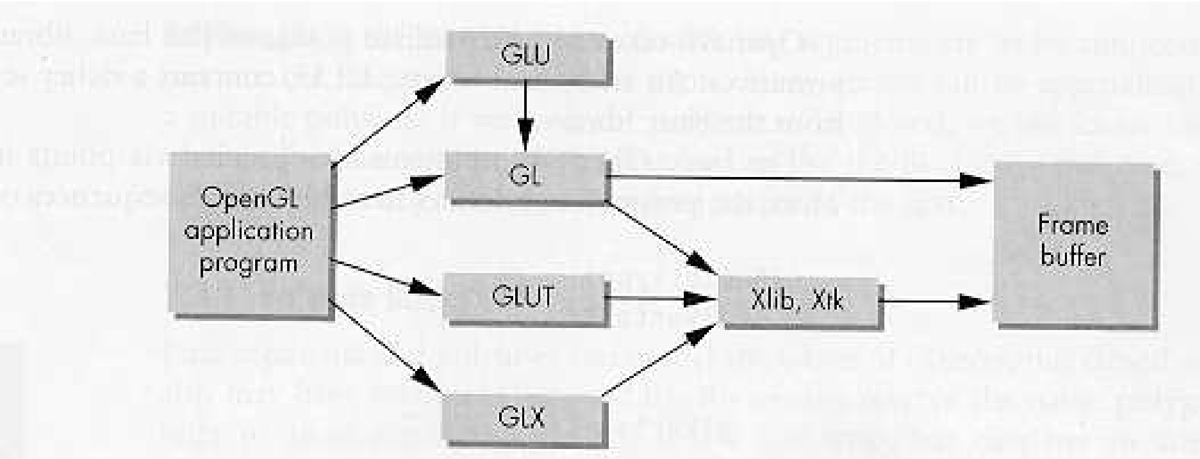
  - OpenGL only performs rendering (drawing) not event handling or window management

  - It consists of three main libraries:

    * *GL* (Graphics Library) - OpenGL functions

    * *GLU* (Graphics Utility Library) - it uses only GL functions but it contains code for common geometric objects, such as spheres, that users prefer not to write repeatedly

    * *GLUT* (Graphics Library Utility Toolkit) - interface between OpenGL and X-windows



- **OpenGL is a state machine**

  - Once you put it in a state, it remains in this state until you **explicitly** change it

  - Examples of state variables: current color, projection transformation, material properties etc.

- **OpenGL Command Syntax**

  - OpenGL function names start with the letters *gl*, *glu* or *glut*

  - OpenGL defined constants start with the letters *GL*, *GLU* or *GLUT*

  - OpenGL function names have the following format:

gl<Command>[234][sifd][v](args ...)

glVertex2f(), *glVertex3f(),* *glVertex3fv()*

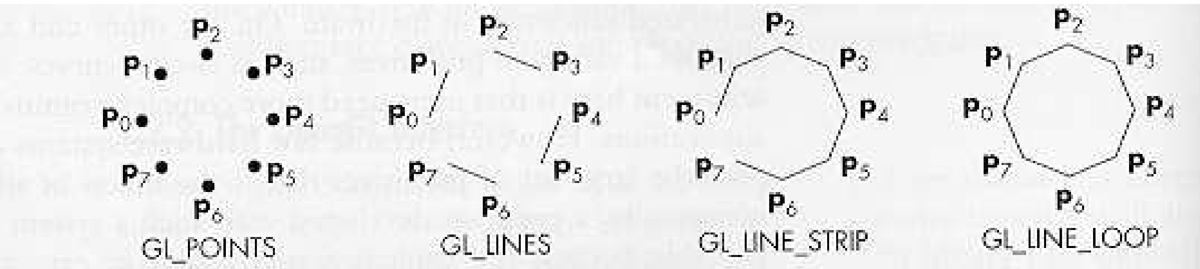
- **OpenGL primitives**

  - OpenGL has a small set of primitives: points, lines, and polygons

  - The basic OpenGL primitives are specified using points or vertices as follows

```
glBegin(type);
  glVertex*(...);
     ...
  glVertex*(...);
glEnd();
```

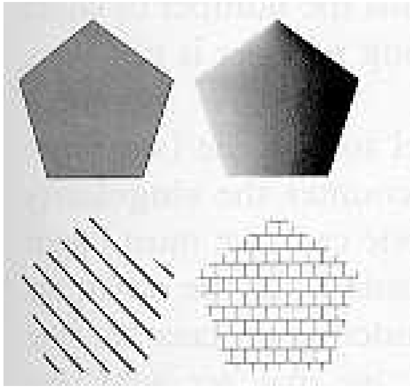
- Line segments



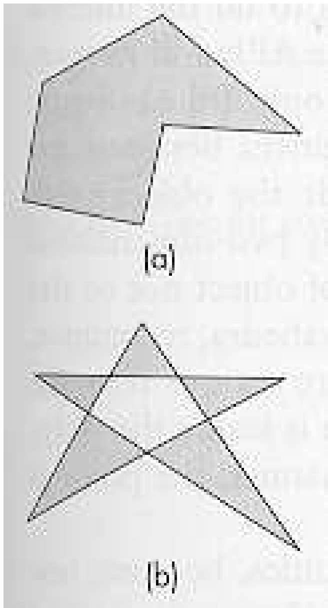
- Polygons

  * The areas enclosed by single closed loops of line segments

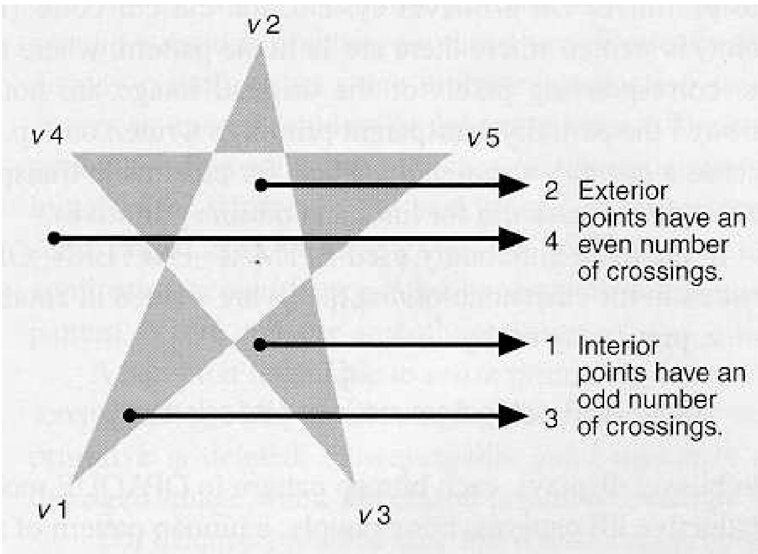
  * Interiors can be filled with solid color or a pattern - edges can or cannot be displayed



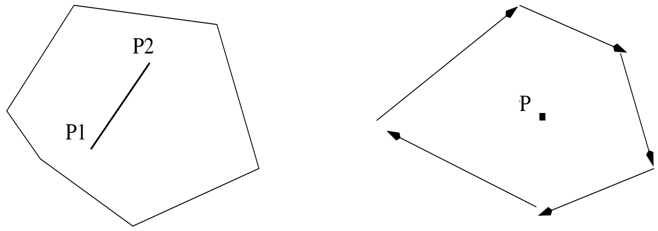
* Filling the interior of simple, non-simple, and convex polygons



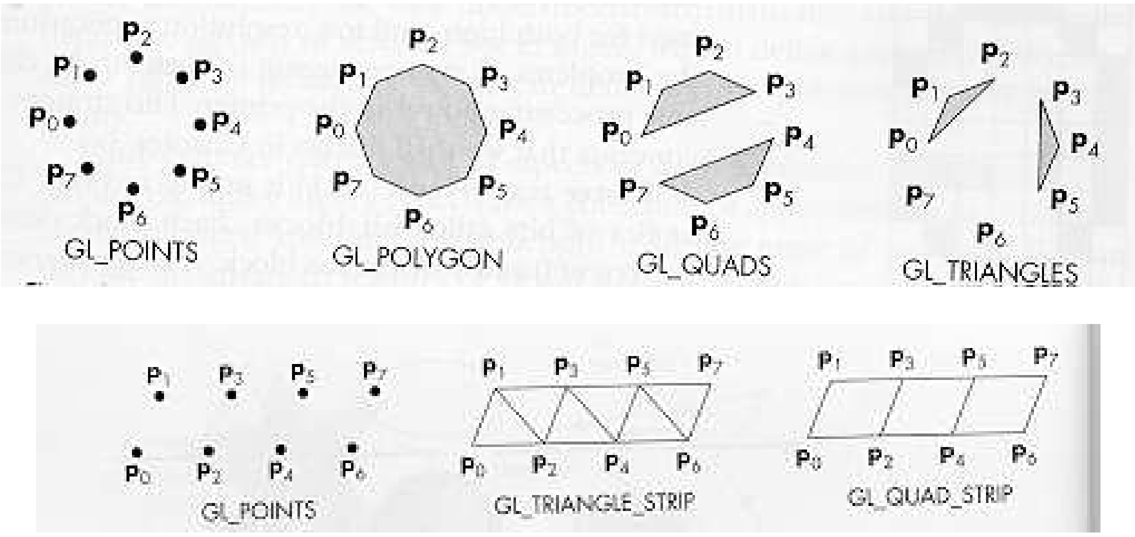
* Testing if a point is in the interior of a polygon



* Testing if a polygon is convex



* Some OpenGL polygon types



- **Changing the state**

  - "On" and "Off" state variables

    * They can be set to one of the following two values:

      GL_TRUE or *FL_FALSE*

    * Some state variables in this category are:

      GL_POINT_SMOOTH, GL_LINE_SMOOTH, GL_LINE_STIPPLE

    * To turn on and off these variables, use the following two functions:

      void glEnable(GLenum pname), void glDisable(GLenum pname)

    * Checking the current status of a variable:

      GLboolean glIsEnabled(GLenum, pname)

- "Mode" state variables

  * They can be set to certain values from a discrete set of values

  * Function calls are specific to the state variables

  * Example: *GL_SHADE_MODEL*

    void glShadeModel(GLenum mode)
    (mode must be in [*GL_SMOOTH, GL_FLAT*])

- "Value" state variables

  * They can be set to certain values from a continuous set of values

  * <u>Color</u>

    void glColor3f(GLfloat red, GLfloat green, GLfloat blue)
    (red, green, blue must be in [0,1])

  * <u>Point size</u>

    void glPointSize(GLfloat size)
    (size must be in [0,1])

  * <u>Line width</u>

    void glLineWidth(GLfloat width)
    (width must be in [0,1])

  * <u>Stippled (dotted or dashed) lines</u>

    void glLineStipple(GLint factor, GLshort pattern)
    (factor must be in [0,255] and pattern is a 16 bit series or 0s and 1s)

  * Checking the current status of a variable:

    *void glGetBooleanv(GLenum pname, GLboolean *params)*
    *void glGetIntegerv(GLenum pname, GLint *params)*
    *void glGetFloatv(GLenum pname, GLfloat *params)*
    *void glGetDoublev(GLenum pname, GLdouble *params)*