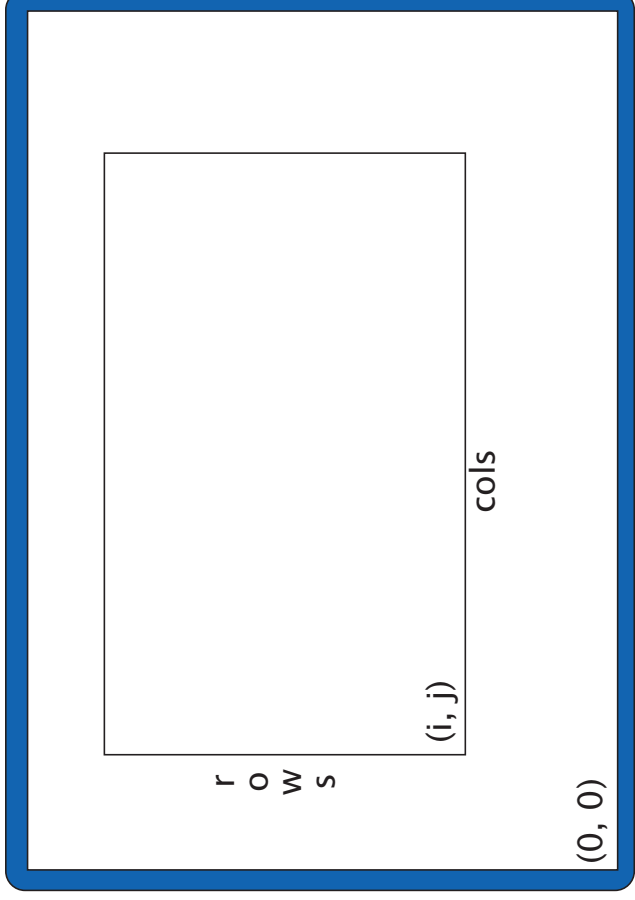


Screen Coordinates

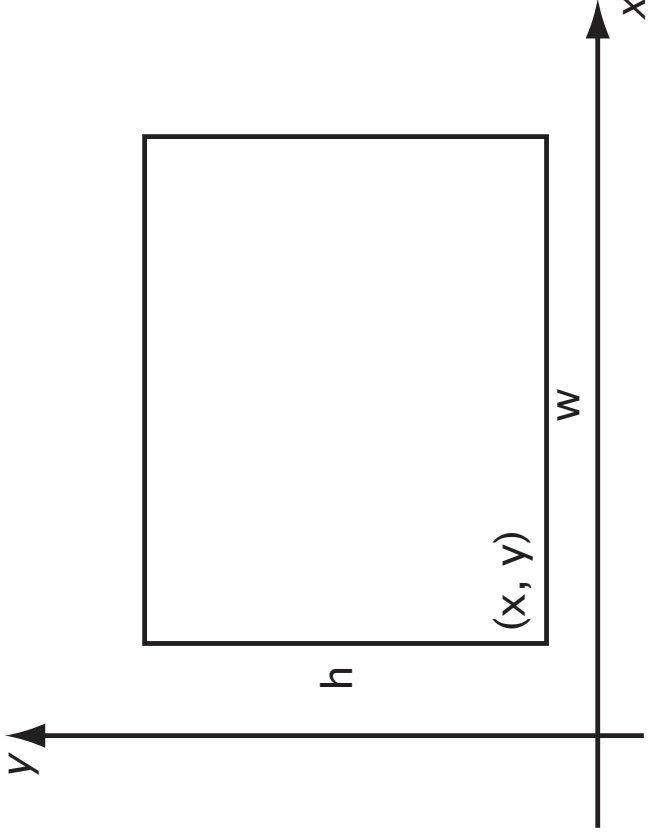


The OpenGL command

```
glViewport(i, j, rows, cols);
```

defines a *viewport*.

World Coordinate

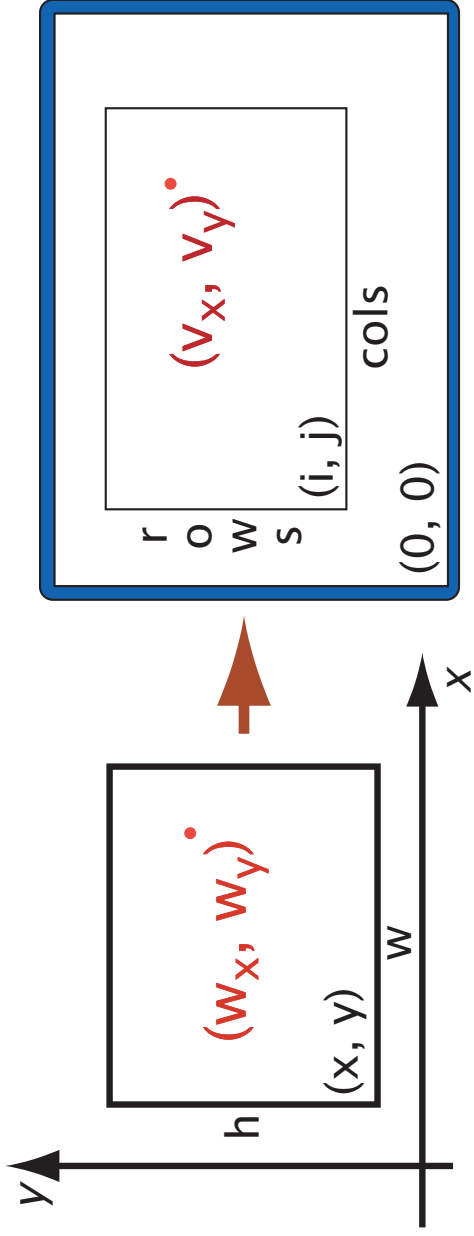


The OpenGL command

```
gluOrtho2D(x, y, x + w, y + h);
```

defines the *window*, and the window-to-viewport transformation.

Window to Viewport Transformation



$$v_x = \frac{cols}{w}(w_x - x)$$

$$v_y = \frac{rows}{h}(w_y - y)$$

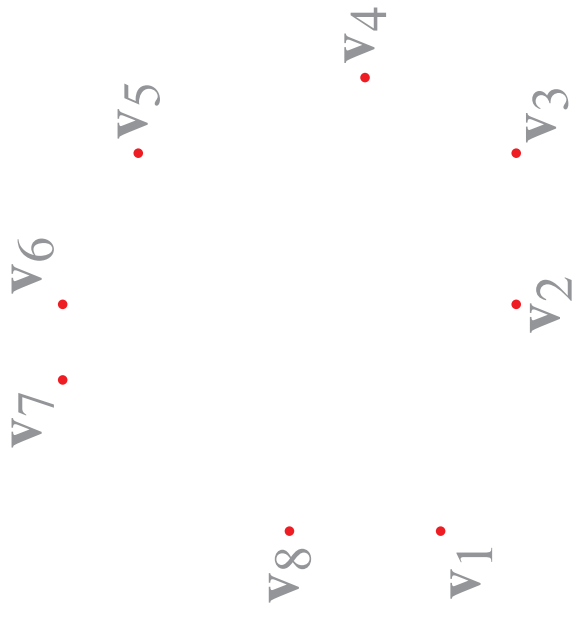
Also called an affine transformation.

OpenGL Drawing Primitives

- Points (`GL_POINTS`)
- Line segments (`GL_LINES`)
- Linear curve (`GL_LINE_STRIP`)
- Open Polygon (`GL_LINE_LOOP`)
- Filled Triangle (`GL_TRIANGLES`)
- Filled Triangle Strip (`GL_TRIANGLE_STRIP`)
- Filled Triangle Fan (`GL_TRIANGLE_FAN`)
- Filled (Convex) Quadrilaterals (`GL_QUADS`)
- Filled (Convex) Quadrilateral Strip (`GL_QUAD_STRIP`)
- Convex Polygon (`GL_POLYGON`)

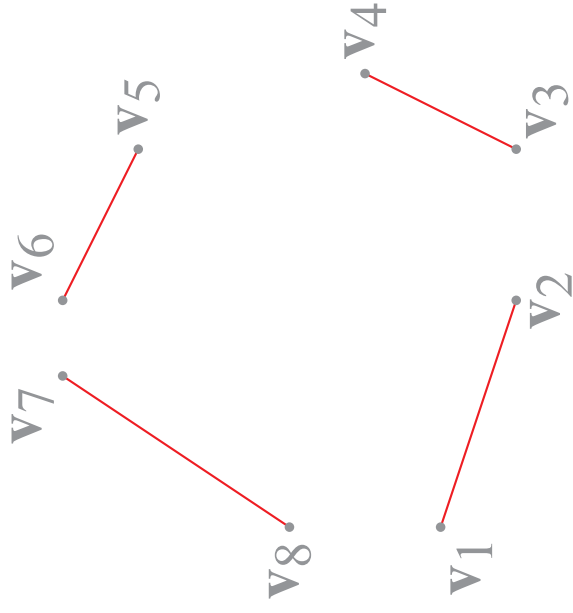
Points

```
glBegin(GL_POINTS);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



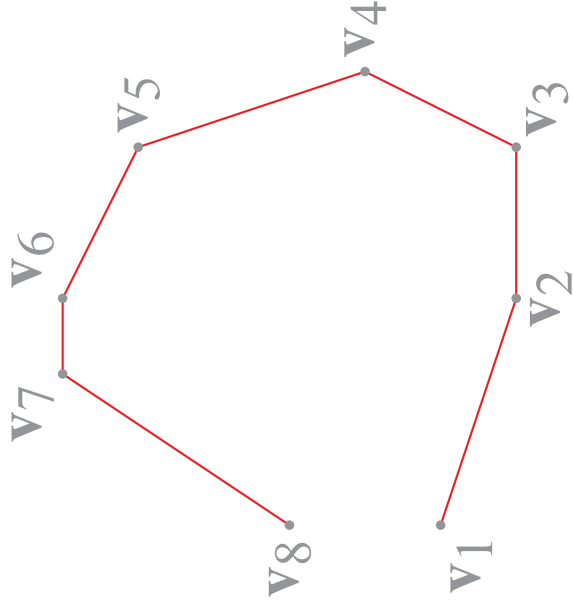
Line Segments

```
glBegin(GL_LINES);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



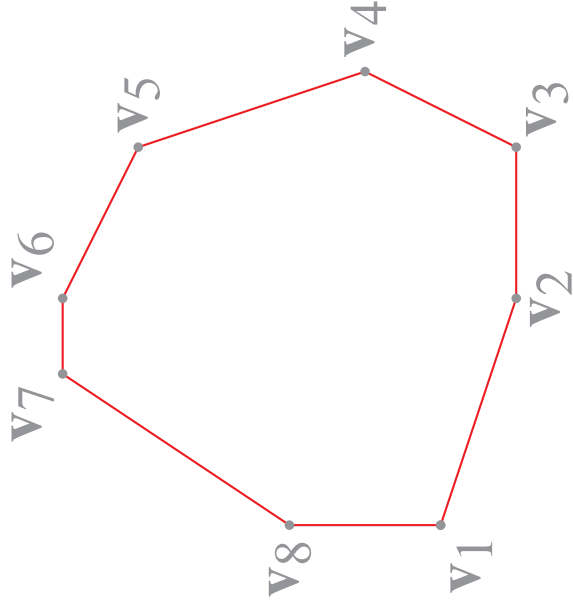
An Open Piecewise-Linear Curve

```
glBegin(GL_LINE_STRIP);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



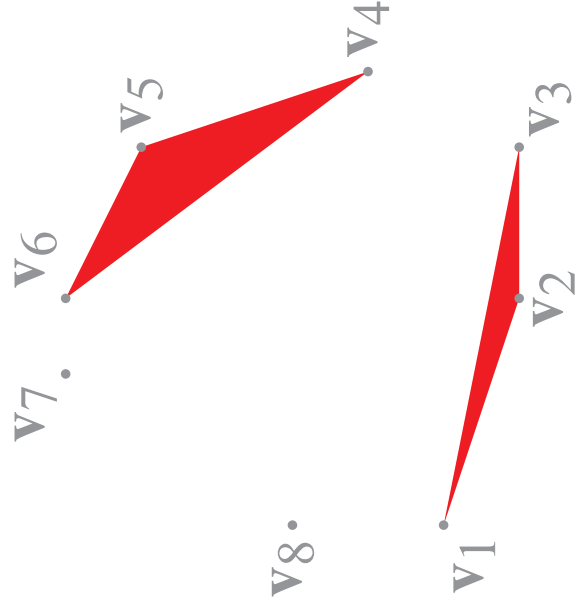
A Closed Piecewise-Linear Curve

```
glBegin(GL_LINE_LOOP);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



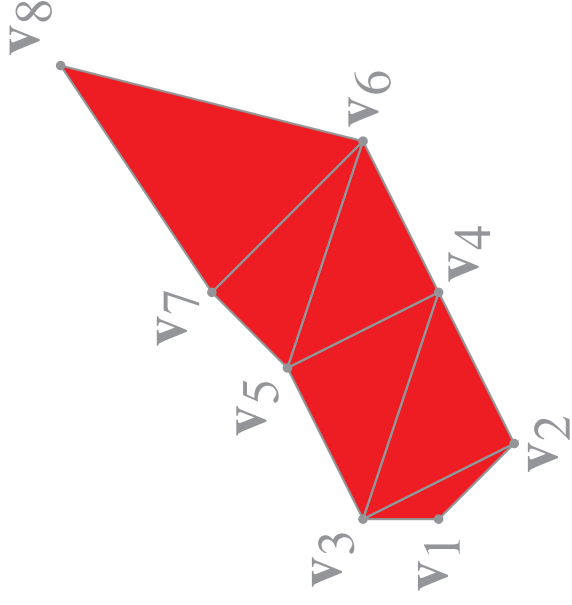
Triangles

```
glBegin(GL_TRIANGLES);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



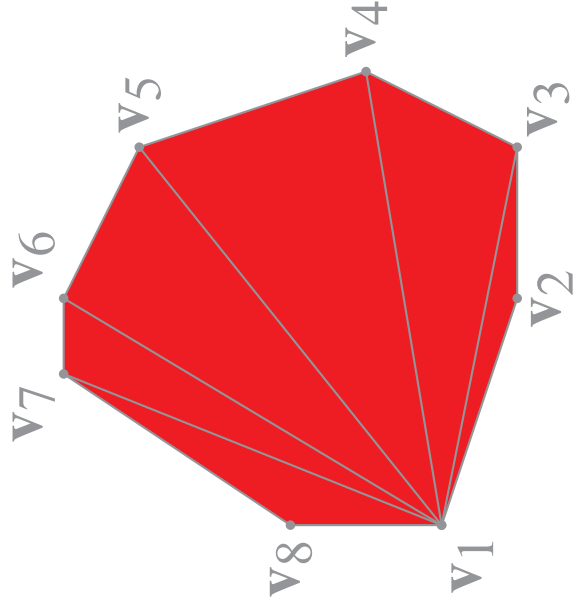
A Triangle Strip

```
glBegin(GL_TRIANGLE_STRIP);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



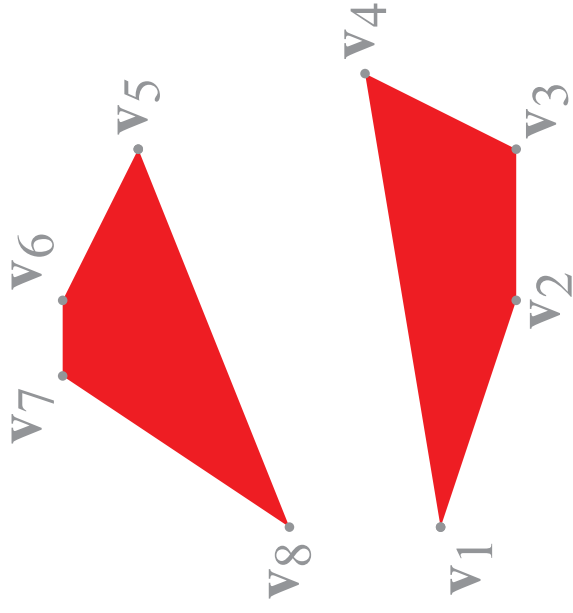
A Triangle Fan

```
glBegin(GL_TRIANGLE_FAN);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



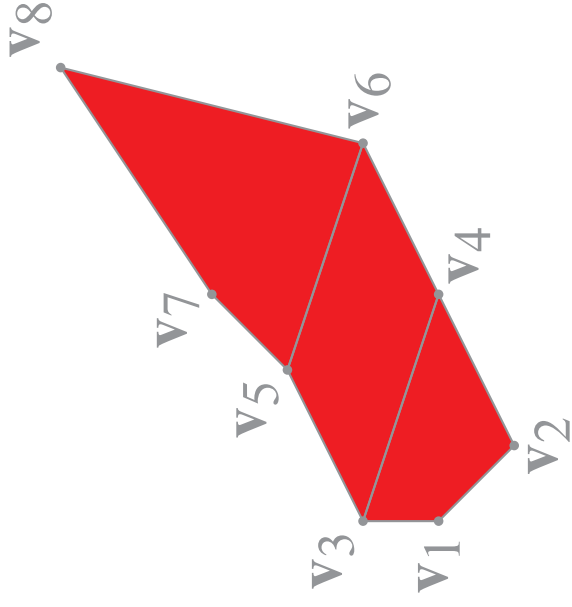
Quadrilaterals

```
glBegin(GL_QUADS);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



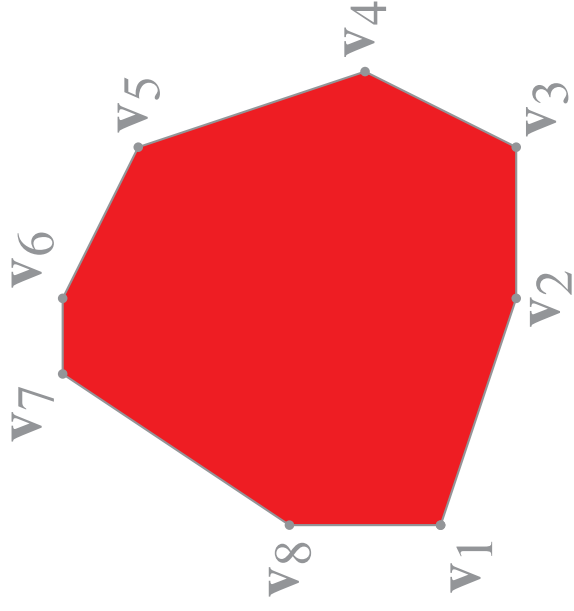
Examples: Quadrilateral Strip

```
glBegin(GL_LINE_LOOP);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



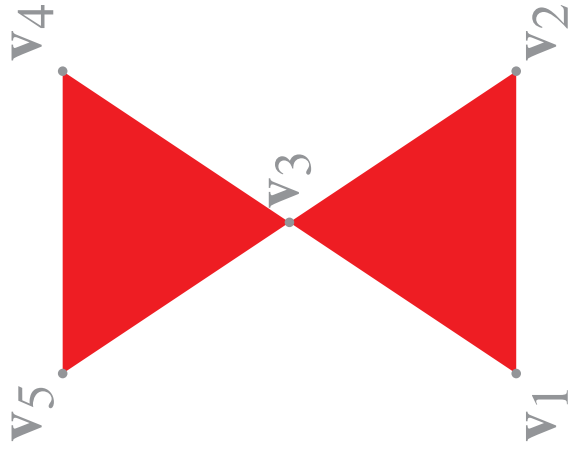
A Convex Polygon

```
glBegin(GL_POLYGON);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v6);  
  glVertex2fv(v7);  
  glVertex2fv(v8);  
glEnd();
```



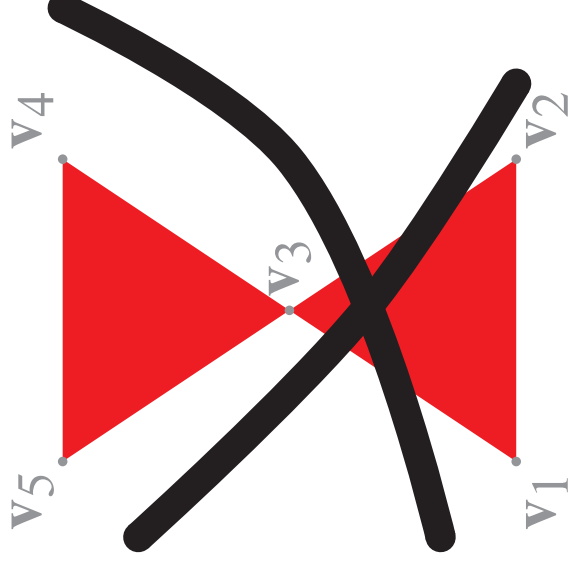
An Hourglass?

```
glBegin(GL_POLYGON);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v3);  
glEnd();
```



An Hourglass? — No! (Nonconvex)

```
glBegin(GL_POLYGON);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v3);  
glEnd();
```



An Hourglass

```
glBegin(GL_TRIANGLES);  
  glVertex2fv(v1);  
  glVertex2fv(v2);  
  glVertex2fv(v3);  
  glVertex2fv(v4);  
  glVertex2fv(v5);  
  glVertex2fv(v3);  
glEnd();
```

