Chapter 2 Solutions

2.9 We can solve this problem separately in the x and y directions. The transformation is linear, that is $x_s = ax + b$, $y_s = cy + d$. We must maintain proportions, so that x_s in the same relative position in the viewport as x is in the window, hence

$$\frac{x - x_{min}}{x_{max} - x_{min}} = \frac{x_s - u}{w},$$

$$x_s = u + w\frac{x - x_{min}}{x_{max} - x_{min}}.$$

Likewise

$$y_s = v + h\frac{x - x_{min}}{y_{max} - y_{min}}.$$

2.11 Problems might arise if the scanline is parallel to one of the sides of the polygon. Scanlines are typically vertical, hence there is no problem with vertical sides. For horizonal sides, we have to decide if the side is inside or outside of the polygon. One solution is to not allow any vertices to exist on a scanline. If we do so, typically by perturning a vertix on a scanline, the horizonal edge problem never arises.

2.13 We could have a list of edges. Each edge could point to a pair of vertices. A polygon could then be represented via a pointer to a list of edges.

2.15 There are two fundametal approaches: vertex lists and edge lists. With vertex lists we store the vertex locations in an array. The mesh is represented as a list of interior polygons (those polygons with no other polygons inside them). Each interior polygon is represented as an array of pointers into the vertex array. To draw the mesh, we traverse the list of interior polygons, drawing each polygon.
One disadvantage of the vertex list is that if we wish to draw the edges in the mesh, by rendering each polygon shared edges are drawn twice. We can avoid this problem by forming an edge list or edge array, each element is a pair of pointers to vertices in the vertex array. Thus, we can draw each edge once by simply traversing the edge list. However, the simple edge list has no information on polygons and thus if we want to render the mesh in some other way such as by filling interior polygons we must add something to this data structure.

A flexible mesh representation would consist of an edge list, a vertex list and a polygon list with pointers so we could know which edges belong to which polygons and which polygons share a given vertex.

2.19 If can display four colors, there are 2 bits per pixel in the frame buffer. Because $64 = 2^{3*2}$, we can display only 4 reds, 4 greens and 4 blues which probably indicates a low quality CRT.