Angel and Shreiner: Interactive Computer Graphics, Eighth Edition

Chapter 2 Solutions

2.2 Each iteration removes the central triangle of the four created by subdivision. If we start with an equilateral triangle, the area is reduced by 3/4 by each iteration. Hence, in the limit we have no area left. If we consider the perimeter, again starting with an equilateral triangle, we find that if the original triangle had a side length of one unit and a perimeter of three units, the three remaining triangles created by the subdivision have a combined perimeter of 4 1/2 units. Thus the perimeter increases at each iteration and in the limit we have an object with no area and an infinite perimeter.

2.8 The subtractive colors are often called the complementary colors. Suppose each color in the RGB system can be written as the triplet (r,g,b) where each value is between 0 and 1. Then, the complement of the color (r,g,b) is the color in CMY space (c,m,y) = (1-r, 1-g, 1-b). If we want a color C in RGB space, we specify its complement in CMY space. For example, if we want a full red, in RGB it is (1,0,0) and its complement is (0,1,1). In CMY, (0,1,1) is magenta and yellow, which have red in common.

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

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

2.10 The biggest advantage of relative positioning is that it corresponds to the way we think. We use terms such as "in front of," "next to,", "to the right," "forward," and "back" in speech to describe the location of objects. For structures that we will consider in Chapter 10, we shall see that when objects have parts such as a model of the human figure, we can describe

their motion much easier in terms of the relative positioning of the components. In implementation, there are additional advantages to relative positioning. For example, if we store the positions of the vertices that determine an object in terms of their relative positions from a particular vertex, we can move the whole object by simply moving the base vertex. The major disadvantage of relative positioning is that if an error is made, all further positions are incorrect.
To support relative positioning we must add some sort of graphics cursor which contains the position we measure from.

2.11 Most practical tests work on a line by line basis. Usually we use scanlines, each of which corresponds to a row of pixels in the frame buffer. If we compute the intersections of the edges of the polygon with a line passing through it, these intersections can be ordered. The first intersection begins a set of points inside the polygon. The second intersection leaves the polygon, the third reenters and so on.

2.12 A simple but inefficient test would be to compute the intersections of all pairs of lines that are determined by the edges of the polygons. We could then test if any of these intersections lie on the edges as opposed to the lines.

2.13 There are two fundamental 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 that gives information as to which edges form each polygon.
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.

2.14 The advantage of an edge list is that when we traverse the edge list, each edge is drawn exactly once which is efficient if polygons are rendered by their edges rather than filled. However, a simple edge list does not contain any polygon by polygon information such as to which polygons a given edge belongs.

2.15 The Maxwell triangle corresponds to the triangle that connects the red, green, and blue vertices in the color cube.

2.18 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.

2.19 Consider the lines defined by the sides of the polygon. We can assign a direction for each of these lines by traversing the vertices in a counter-clockwise order. One very simple test is obtained by noting that any point inside the object is on the left of each of these lines. Thus, if we substitute the point into the equation for each of the lines (ax+by+c), we should always get the same sign.

2.21 Each of the four tetrahedrons that are created has 1/8 of the volume of the original tetrahedron. Hence, by keeping only these four and removing the middle volume, the resulting volume is half the original volume. Each of the triangles that we create when we subdivide has 1/4 the area of the original face. Thus each subtetrahedron has the same area as one of the original faces and in total the surface area of the four subtetrahedrons is the same as the surface area of the original tetrahedron, even though the central section has been removed. If we repeat the calculation for the length of all edges, we find the length of all edges kept after a subdivision is greater the the length of the edges of the original tetrahedron.