Computer Graphics

Lecture 12

Area-Fill Attributes: Options for filling a defined region include a choice between a solid colour or a patterned fill and choices for the particular colours and patterns. These fill options can be applied to polygon regions or to areas defined with curved boundaries, depending on the capabilities of the available package. In addition, areas can be painted using various brush styles, colours, and transparency parameters.

Fill Styles: Areas are displayed with three basic fill styles: hollow with a colour border, filled with a solid colour, or Wed with a specified pattern or design. A basic fill style is selected in a PHIGS program with the function

setInteriorStyle(fs)

Values for the fill-style parameter f s include hollow, solid, and pattern (Fig. 4-18). Another value for fill style is hatch, which is used to fill an area with selected hatching patterns-parallel lines or crossed lines--as in Fig. 4-19. As with line attributes, a selected fill-style value is recorded in the list of system attributes and applied to fill the interiors of subsequently specified areas. Fill selections for parameter fs are normally applied to polygon areas, but they can also be implemented to fill regions with curved boundaries

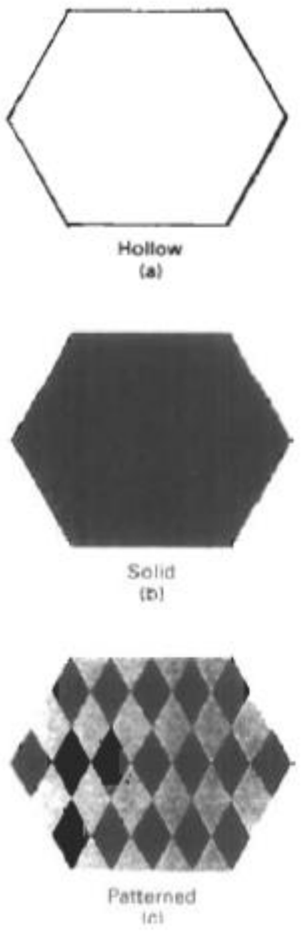

Figure 4-18
Polygon fill styles.

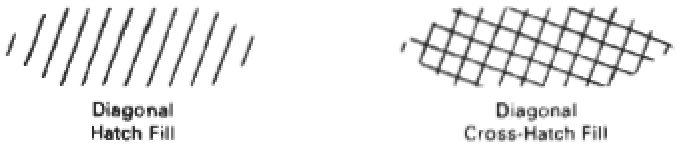

Figure 4-19
Polygon fill using hatch patterns.

Hollow areas are displayed using only the boundary outline, with the interior colour the same as the background colour. A solid fill is displayed in a single colour up to and including the borders of the region. The colour for a solid interior or for a hollow area outline is chosen with

setInteriorColourIndex(fc)

where fill colour parameter fc is set to the desired colour code. A polygon hollow fill is generated with a line drawing routine as a closed polyline.

Solid fill of a region can be accomplished with the scan-line procedures discussed previously.

Other fill options include specifications for the edge type, edge width, and edge colour of a region. These attributes are set independently of the fill style or fill colour, and they provide for the same options as the line-attribute parameters (line type, line width, and line colour). That is, we can display area edges dotted or dashed, fat or thin, and in any available colour regardless of how we have filled the interior.

Pattern Fill: We select fill patterns with

setInteriorStyleIndex (pi)

where pattern index parameter pi specifies a table position. For example, the following set of statements would fill the area defined in the *fillArea* command with the second pattern type stored in the pattern table

CHARACTER ATTRIBUTES: The appearance of displayed characters is controlled by attributes such as font, size, colour, and orientation. Attributes can be set both for entire character strings (text) and for individual characters defined as marker symbols

Text Attributes: There are a great many text options that can be made available to graphics programmers. First of all, there is the choice of font (or typeface), which is a set of characters with a particular design style such as New York, Courier, Helvetica, London, 'Times Roman, and various special symbol groups. The characters in a selected font can also be displayed with assorted underlining styles (solid, dotted, double), in boldface, in italics, and in outline or shadow styles. A particular font and associated stvle is selected in a PHIGS program by setting an integer code for the text font parameter tf in the function

setTextFont (tf)

Font options can be made available as predefined sets of grid patterns or as character sets designed with polylines and spline curves

Colour settings for displayed text are stored m the system attribute list and used by the procedures that load character definitions into the frame buffer. When a character string is to be displayed, the current colour is used to set pixel values in the frame buffer corresponding to the character shapes and positions. Control of text colour (or intensity) is managed from an application program with

setTextColour (tc)

where text colour parameter tc specifies an allowable colour code

Marker Attribute: A marker symbol is a single character that can he displayed in different colours and in different sizes. Marker attributes are implemented by procedures that load the chosen character into the raster at the defined positions with the specified colour and size

We select a particular character to be the marker symbol with

setMarkerType (mt)

where marker type parameter mt is set to an integer code. Typical codes for marker type are the integers 1 through 5, specifying, respectively, a dot (.) , a vertical cross (+), an asterisk(*), a circle, and a diagonal cross. Displayed marker types are centred on the marker coordinates.

We set the marker size with parameter marker size ms assigned a positive number

setMarkerSizeScaleFactor (ms)