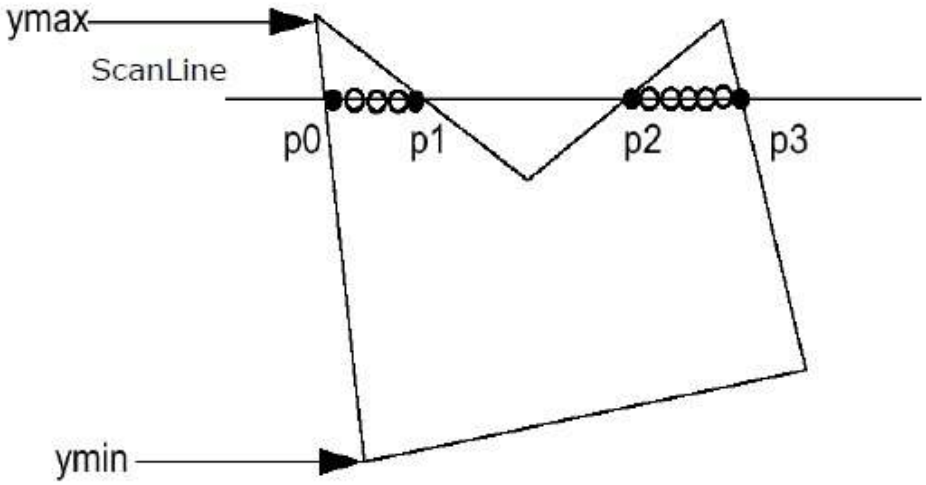
What is Polygon Filling Algorithm?

An ordered list of vertices forms a polygon. The pixels that fall on the border of the polygon are determined and the pixels that fall inside are determined in order to colour the polygon.

Scan Line Algorithm

The polygon edges are being intersected with the scanline by Scan Line Algorithm. The polygon is filled with colours in between the intersecting pairs. The practical working of the algorithm is shown below:

Step 1 − Find out the Ymin and Ymax from the given polygon.



Step 2 – From each edge of the polygon from Ymin to Ymax, all the egdes are intersected by Scanline. Each of the points of intersection are named as p0, p1, p2, p3.

Step 3 − Sort the intersection point in the increasing order of X coordinate i.e. (p0, p1), (p1, p2), and (p2, p3).

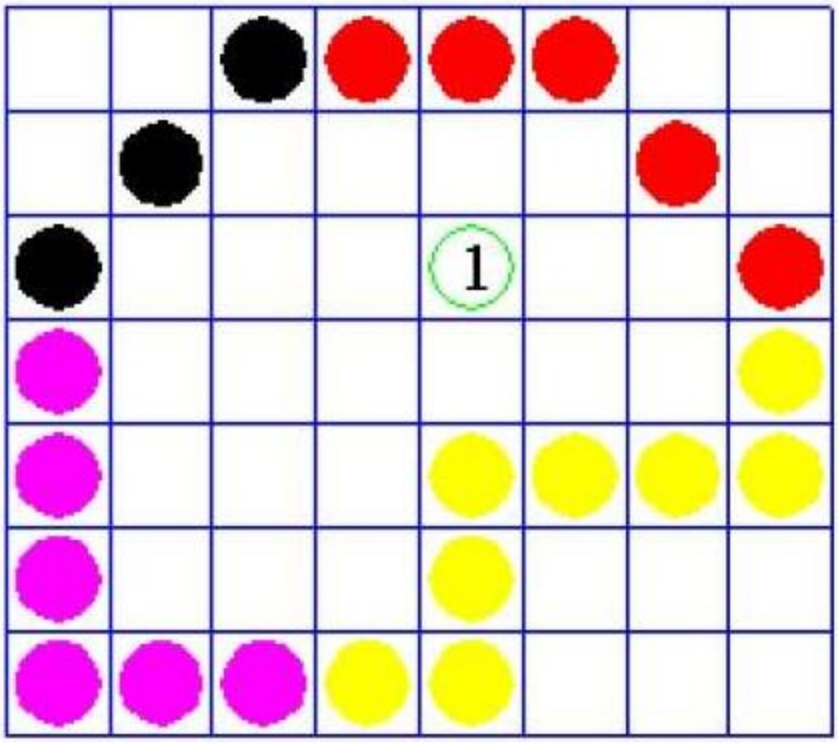
Step 4 − Fill all those pair of coordinates that are inside polygons and ignore the alternate pairs.

Flood Fill Algorithm

For some of the polygons, the area and boundary is filled by using different colours. A specific interior colour is used in such cases.

Fill colour option is used rather than on the object boundary. Fill clour replaces the interior colour. The algorithm is said to be complete when there are no left out pixels of the original interior colour.

The pixels are filled by using either Four-connect or Eight-connect method. The adjacent pixels are considered.



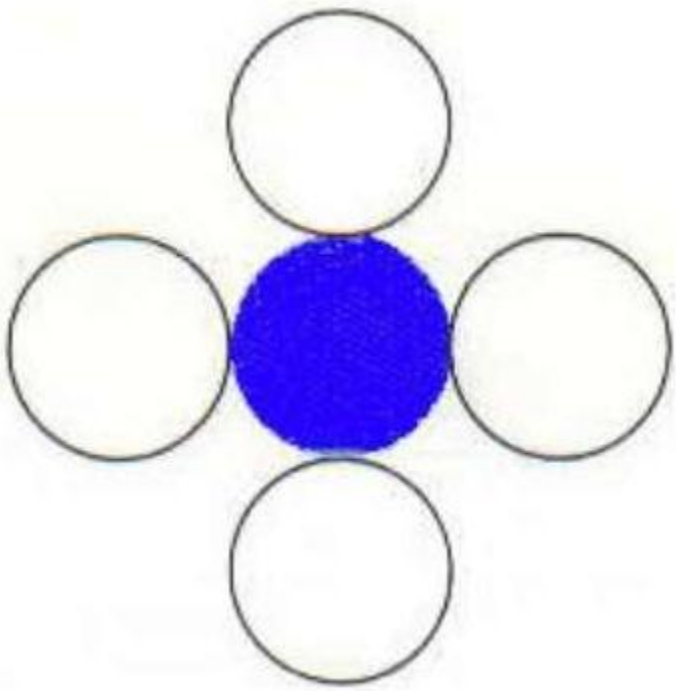
Boundary Fill Algorithm

As the name implies, this algorithm works. A point inside an object is picked and is filled until the boundary is hit by the object. This algorithm works only when the colour of the boundary is different from the colour that is used for filling.

For the complete object, the boundary colour is assumed to be the same. Either 4-connected pixels or 8-connected pixels are used by this algorithm.

4-Connected Polygon

The pixels are used by placing them on four different sides of the current pixel till a different colour boundary is identified in 4-connected polygon.



Algorithm

Step 1 − Initialize the value of seed point (seedx, seedy), fcolor and dcol.

Step 2 − Define the boundary values of the polygon.

Step 3 − Check if the current seed point is of default color, then repeat the steps 4 and 5 till the boundary pixels reached.

If getpixel(x, y) = dcol then repeat step 4 and 5

Step 4 − Change the default color with the fill color at the seed point.

setPixel(seedx, seedy, fcol)

Step 5 − Recursively follow the procedure with four neighborhood points.

FloodFill (seedx – 1, seedy, fcol, dcol)

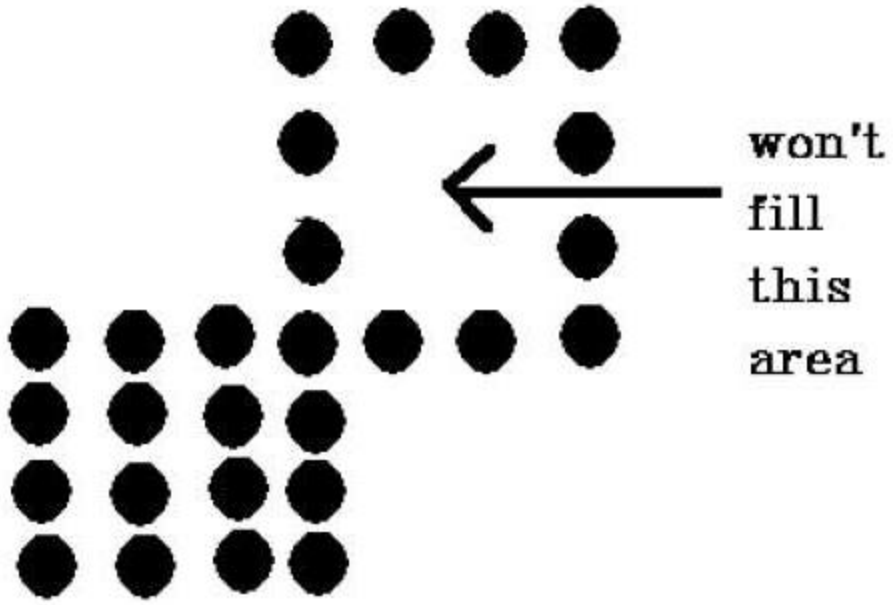
FloodFill (seedx + 1, seedy, fcol, dcol)

FloodFill (seedx, seedy - 1, fcol, dcol)

FloodFill (seedx – 1, seedy + 1, fcol, dcol)

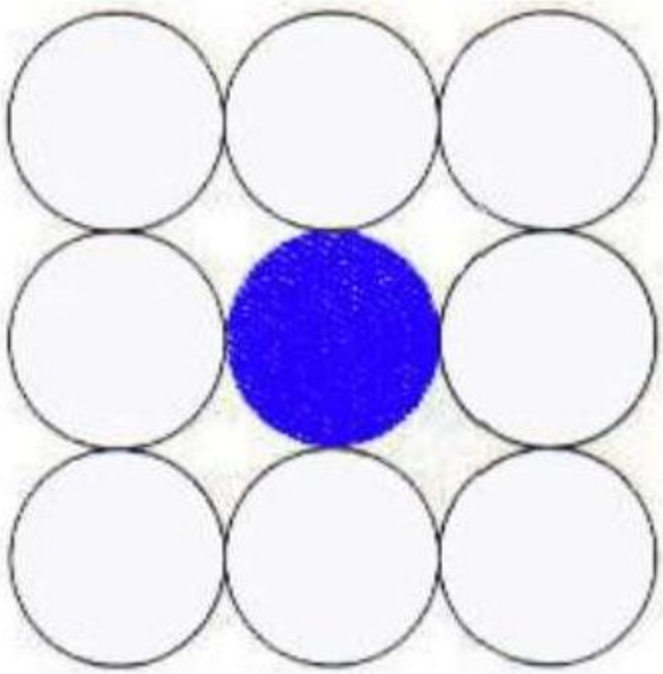
Step 6 − Exit

This technique is linked with a problem. In the case shown below, the image is partially filled, in such instances; only 4-connected pixel techniques cannot be used.



8-Connected Polygon

8-connected pixels are used in this technique by adding the pixels above, below, right and left side of the current pixels. In addition to this, the pixels are fixed diagonally such that the complete area of the current pixel is covered. The process is continued till a boundary is found with different colour.



Algorithm

Step 1 − Initialize the value of seed point (seedx, seedy), fcolor and dcol.

Step 2 − Define the boundary values of the polygon.

Step 3 − Check if the current seed point is of default color then repeat the steps 4 and 5 till the boundary pixels reached

If getpixel(x,y) = dcol then repeat step 4 and 5

2

Step 4 − Change the default color with the fill color at the seed point.

setPixel(seedx, seedy, fcol)

3

Step 5 – With the four neighbourhood points, follow the procedure:

FloodFill (seedx – 1, seedy, fcol, dcol)

FloodFill (seedx + 1, seedy, fcol, dcol)

FloodFill (seedx, seedy - 1, fcol, dcol)

FloodFill (seedx, seedy + 1, fcol, dcol)

FloodFill (seedx – 1, seedy + 1, fcol, dcol)

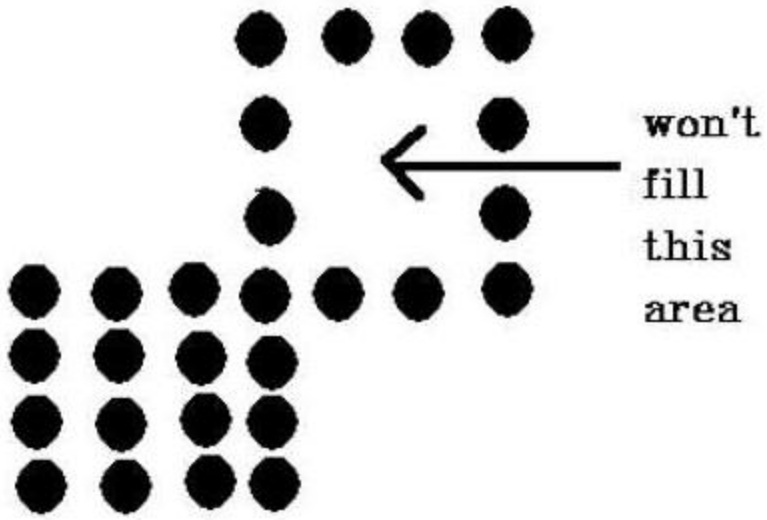
FloodFill (seedx + 1, seedy + 1, fcol, dcol)

FloodFill (seedx + 1, seedy - 1, fcol, dcol)

FloodFill (seedx – 1, seedy - 1, fcol, dcol)

Step 6 − Exit

The area marked in the figure, which 4-connected pixel technique failed to fill is filled by the 8-connected technique.



Inside-outside Test

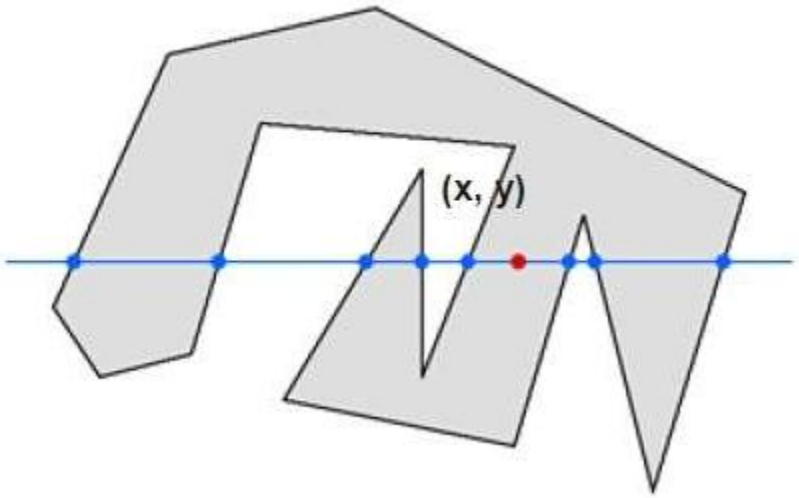
When an object is filled, it is essential to identify whether the specific point is inside the object or outside the object. This is done by Inside-outside test also known as counting number method. An object can be identified whether inside or outside by two methods:

- Odd-Even Rule

- Nonzero winding number rule

Odd-Even Rule

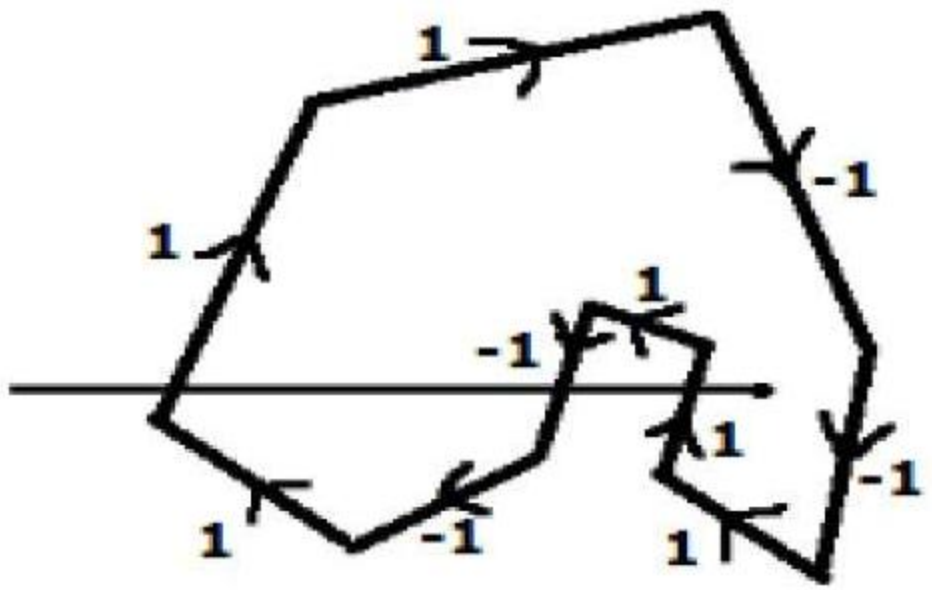
The edge that crosses along the line from the point (x,y) to infinity is counted. In case of odd interactions, the point (x,y) is an interior point and in case of even interactions, the point (x,y) is an exterior point. This concept is depicted by the following example:



It is observed from the above point (x,y), that the number of interaction point on the left side is 5 and on the right side is 3. The point is considered within the object as the number of interaction points is considered as odd.

Nonzero Winding Number Rule

In order to test whether the point is interior or not, simple polygons are used which can be understood by using a pin and a rubber band. Pin is fixed on one of the edges of polygon and the rubber band is tied to it, and by stretching the rubber band along with the edges of the polygon.



Alternatively, directions can be given to all the edges of the polygon. A scan line can be drawn from the point to be tested towards the left most direction of X direction.

- To all the edges going to upward direction the value of 1 is given and for other -1 is assigned as direction values.
- The edge direction values are checked from which the scan line is passing and sum up them.
- The point that is to be tested is an interior point, if the total sum of the direction value is non-zero, or else it is an exterior point.
- The direction values from which the scan line is passed is summed up in the figure above, and then the total would be 1 – 1 + 1 = 1, which is non-zero and hence the point is an interior point.