

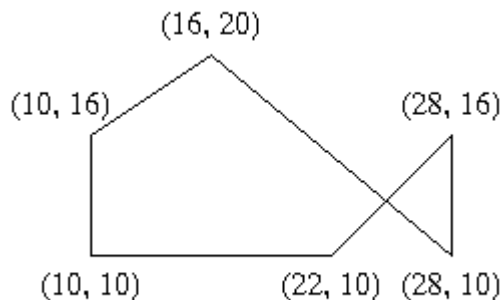
## A Simple Example of Polygon Filling

Just to reiterate the algorithm, the following simple example of scan-line polygon filling will be outlined.

Initially, each vertex of the polygon is given in the form of (x,y) and is in an ordered array as such:

ordered_vertices	
0	(10, 10)
1	(10, 16)
2	(16, 20)
3	(28, 10)
4	(28, 16)
5	(22, 10)

Unfilled, the polygon would look like this to the human eye:



We will now walk through the steps of the algorithm to fill in the polygon.

### 1. Initializing All of the Edges:

We want to determine the minimum y value, maximum y value, x value, and  $1/m$  for each edge and keep them in the all\_edges table. We determine these values for the first edge as follows:

**Y-min:**

Since the first edge consists of the first and second vertex in the array, we use the y values of those vertices to choose the lesser y value. In this case it is 10.

**Y-max:**

In the first edge, the greatest y value is 16.

**X-val:**

Since the x value associated with the vertex with the highest y value is 10, 10 is the x value for this edge.

**1/m:**

Using the given formula, we get  $(10-10)/(16-10)$  for 1/m.

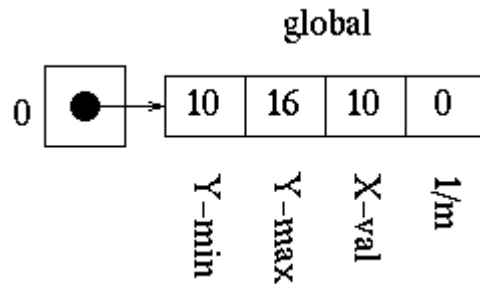
The edge value results are in the form of Y-min, Y-max, X-val, Slope for each edge array pointed to in the all\_edges table. As a result of calculating all edge values, we get the following in the all\_edges table.

all_edges						
0	●	→	10	16	10	0
1	●	→	16	20	10	1.5
2	●	→	10	20	28	-1.2
3	●	→	10	16	28	0
4	●	→	10	16	22	1
5	●	→	10	10	10	inf
			Y-min	Y-max	X-val	1/m

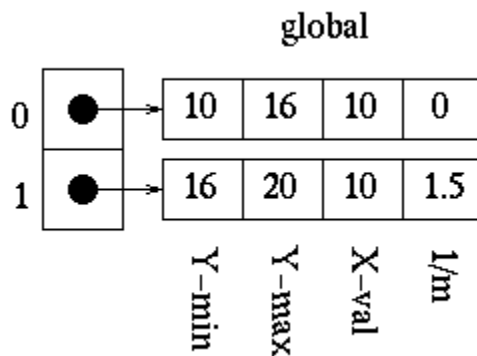
**2. Initializing the Global Edge Table:**

We want to place all the edges in the global edge table in increasing y and x values, as long as slope is not equal to zero.

For the first edge, the slope is not zero so it is placed in the global edge table at index=0.



For the second edge, the slope is not zero and the minimum y value is greater than that at zero, so it is placed in the global edge table at index=1.



For the third edge, the slope is not zero and the minimum y value is equal the edge's at index zero and the x value is greater than that at index 0, so the index is increased to 1. Since the third edge has a lesser minimum y value than the edge at index 2 of the global edge table, the index for the third edge is not increased again. The third edge is placed in the global edge table at index=1.

global						
0	●	→	10	16	10	0
1	●	→	10	20	28	-1.2
2	●	→	16	20	10	1.5
			Y-min	Y-max	X-val	1/m

We continue this process until we have the following:

global						
0	●	→	10	16	10	0
1	●	→	10	16	22	1
2	●	→	10	16	28	0
3	●	→	10	20	28	-1.2
4	●	→	16	20	10	1.5
			Y-min	Y-max	X-val	1/m

Notice that the global edge table has only five edges and the all\_edges table has six. This is due to the fact that the last edge has a slope of zero and, therefore, is not placed in the global edge table.

### 3. Initializing Parity

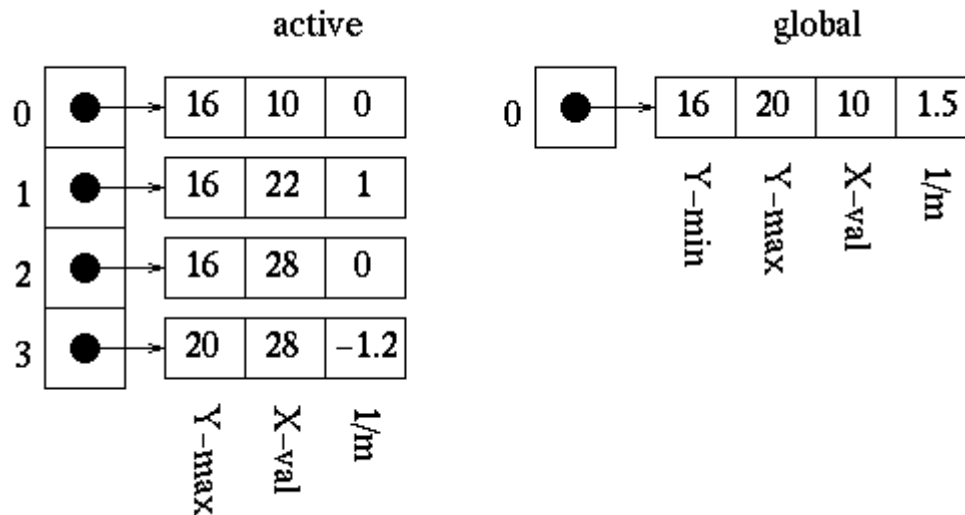
Parity is initially set to even.

### 4. Initializing the Scan-Line

Since the lowest y value in the global edge table is 10, we can safely choose 10 as our initial scan-line.

## 5. Initializing the Active Edge Table

Since our scan-line value is 10, we choose all edges which have a minimum y value of 10 to move to our active edge table. This results in the following.



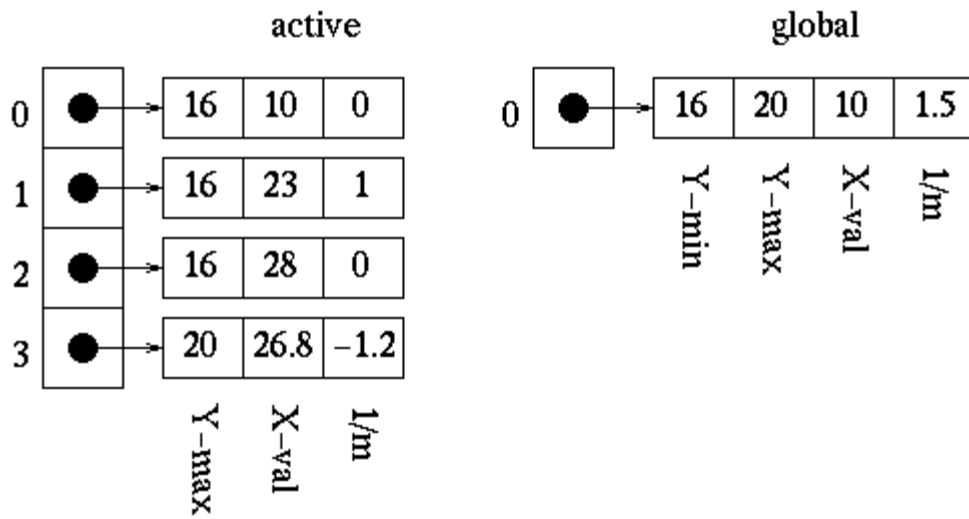
## 6. Filling the Polygon

Starting at the point (0,10), which is on our scan-line and outside of the polygon, will want to decide which points to draw for each scan-line.

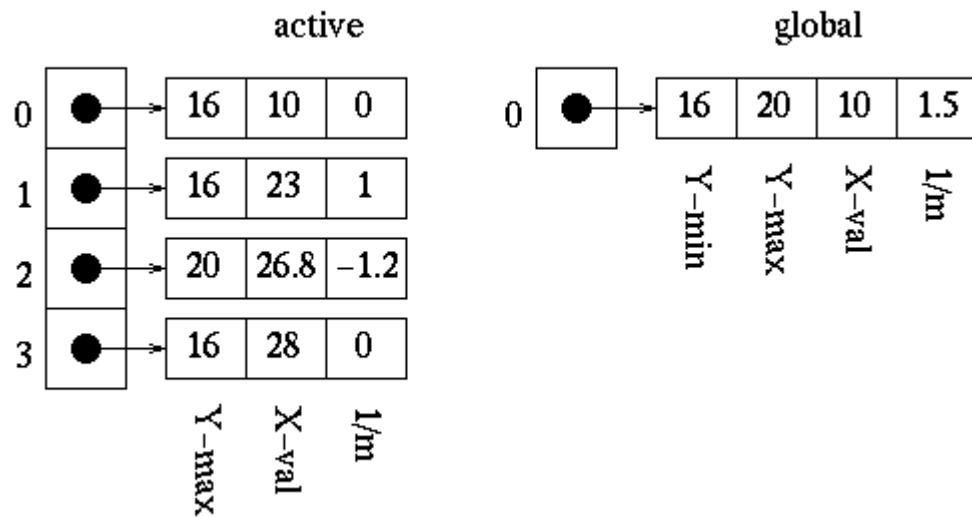
### 1. Scan-line = 10:

Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=22$ . Parity is then changed to even. The next edge is reached at  $x=28$ , and the point is drawn once on this scan-line due to the special parity case. We are now done with this scan-line.

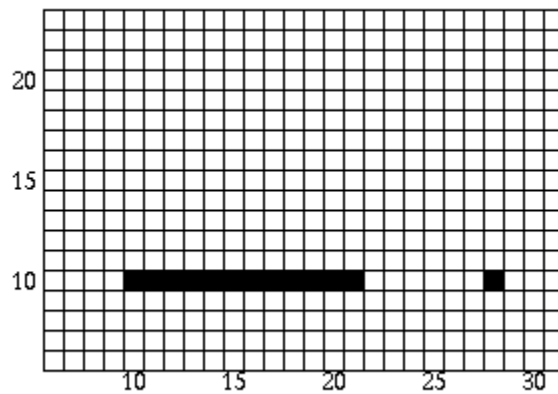
First, we update the x values in the active edge table using the formula  $x_1 = x_0 + 1/m$  to get the following:



The edges then need to be reordered since the edge at index 3 of the active edge table has a lesser x value than that of the edge at index 2. Upon reordering, we get:



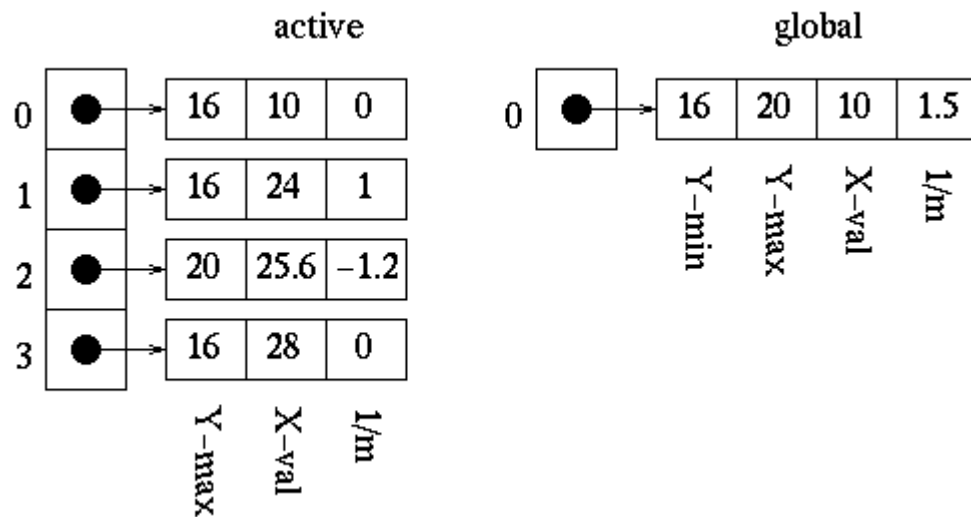
The polygon is now filled as follows:



## 2. Scan-line = 11:

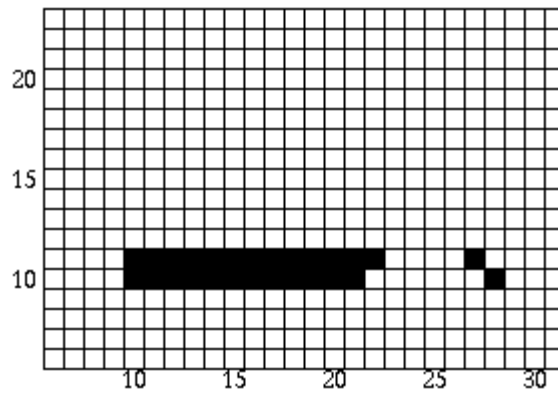
Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=23$ . Parity is then changed to even. The next edge is reached at  $x=27$  and parity is changed to odd. The points are then drawn until the next edge is reached at  $x=28$ . We are now done with this scan-line.

Upon updating the x values, the edge tables are as follows:



It can be seen that no reordering of edges is needed at this time.

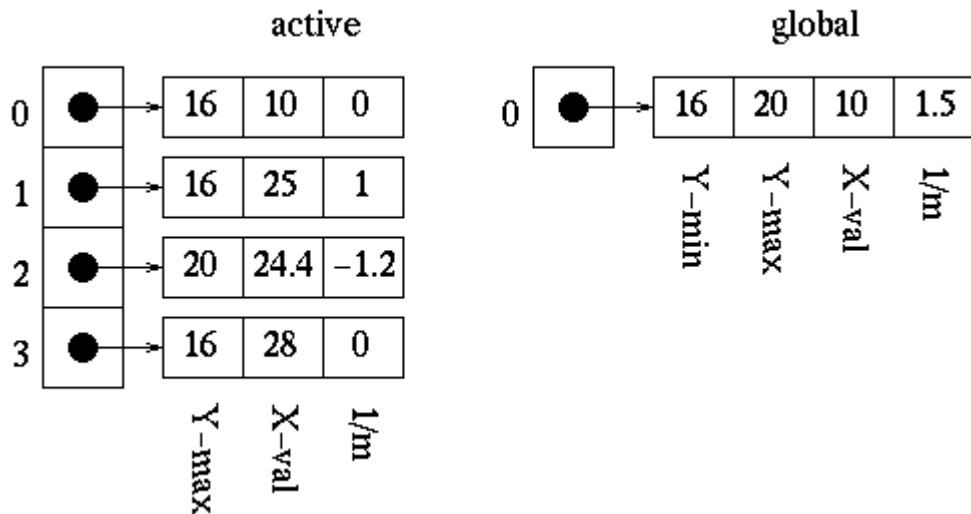
The polygon is now filled as follows:



3. Scan-line = 12:

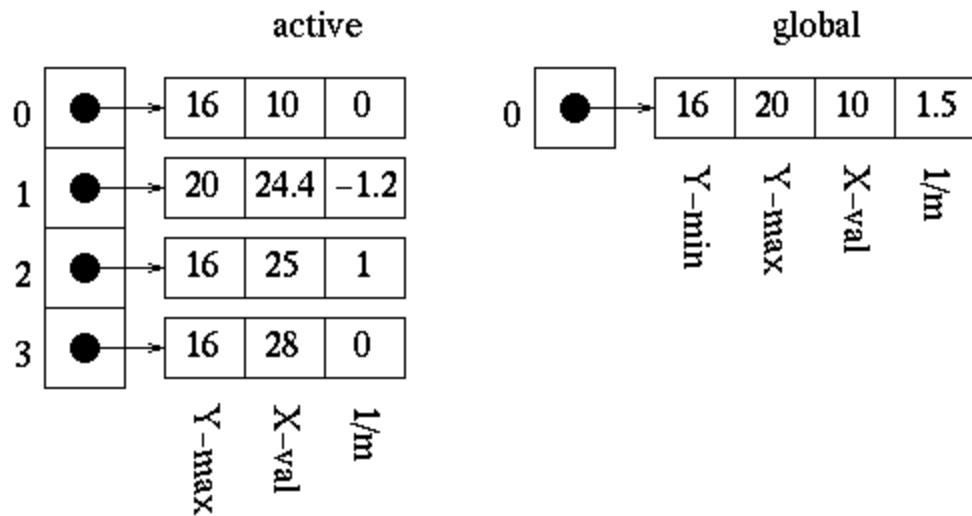
Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=24$ . Parity is then changed to even. The next edge is reached at  $x=26$  and parity is changed to odd. The points are then drawn until the next edge is reached at  $x=28$ . We are now done with this scan-line.

Updating the x values in the active edge table gives us:

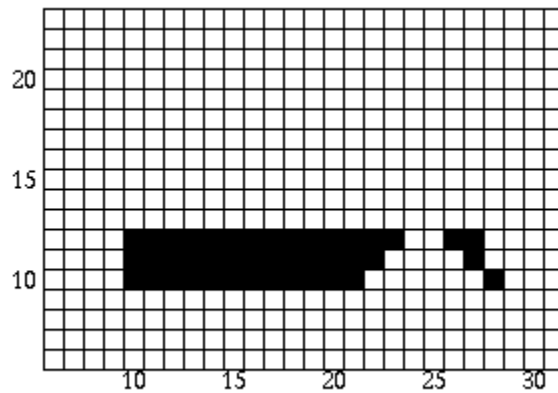


We can see that the active edges need to be reordered since the x value of 24.4 at index 2 is less than the x value of 25 at index 1. Reordering produces the following:





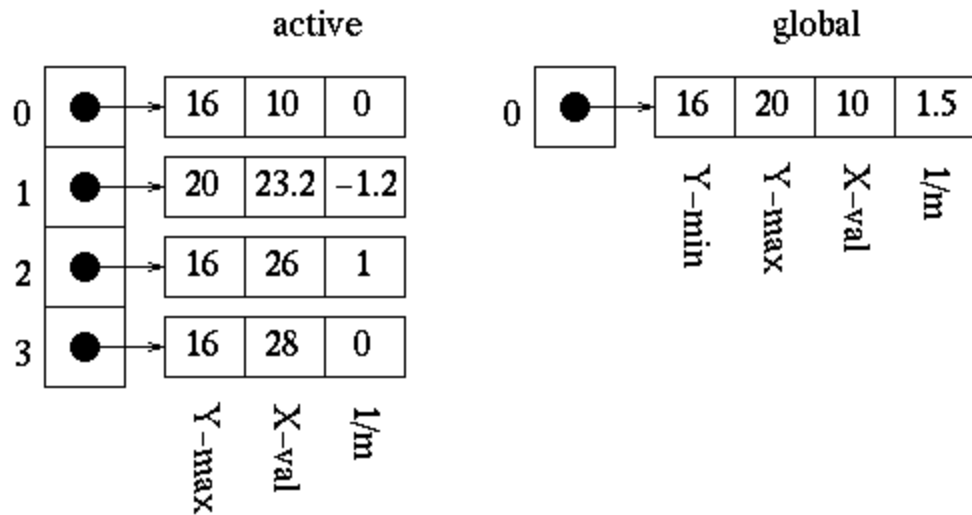
The polygon is now filled as follows:



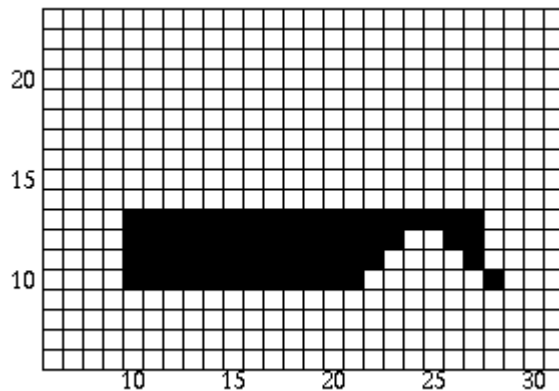
#### 4. Scan-line = 13:

Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=25$ . Parity is then changed to even. The next edge is reached at  $x=25$  and parity is changed to odd. The points are then drawn until the next edge is reached at  $x=28$ . We are now done with this scan-line.

Upon updating the  $x$  values for the active edge table, we can see that the edges do not need to be reordered.



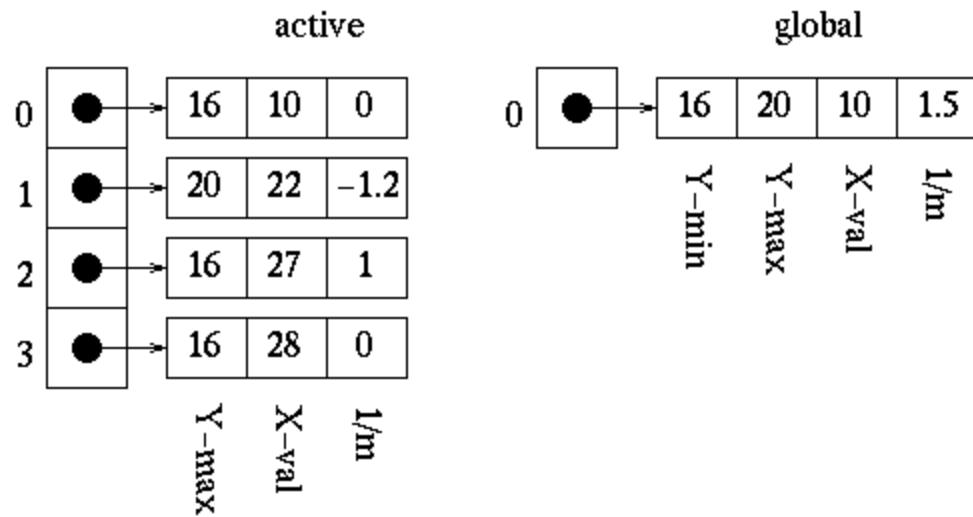
The polygon is now filled as follows:



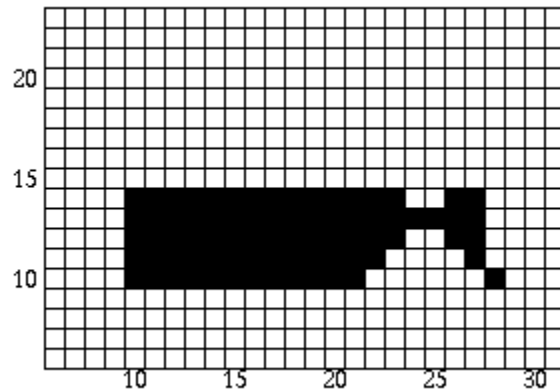
5. Scan-line = 14:

Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=24$ . Parity is then changed to even. The next edge is reached at  $x=26$  and parity is changed to odd. The points are then drawn until the next edge is reached at  $x=28$ . We are now done with this scan-line.

Upon updating the  $x$  values for the active edge table, we can see that the edges still do not need to be reordered.



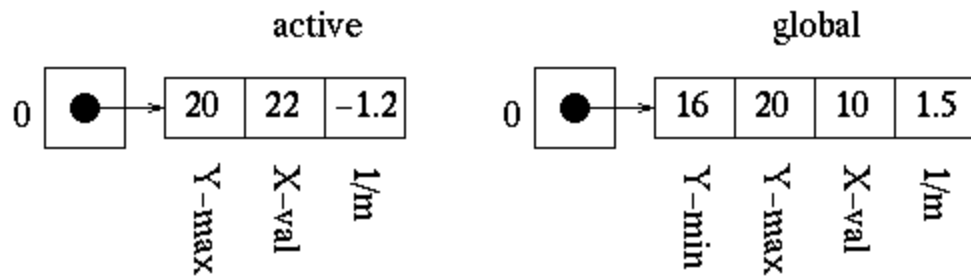
The polygon is now filled as follows:



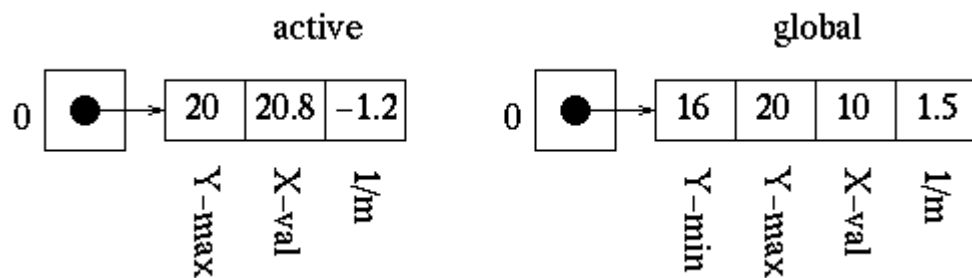
6. Scan-line = 15:

Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is encountered at  $x=22$ . Parity is then changed to even. The next edge is reached at  $x=27$  and parity is changed to odd. The points are then drawn until the next edge is reached at  $x=28$ . We are now done with this scan-line.

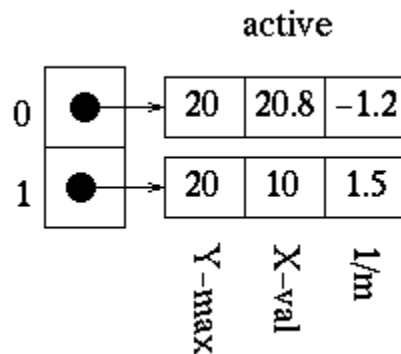
Since the maximum  $y$  value is equal to the next scan-line for the edges at indices 0, 2, and 3, we remove them from the active edge table. This leaves us with the following:



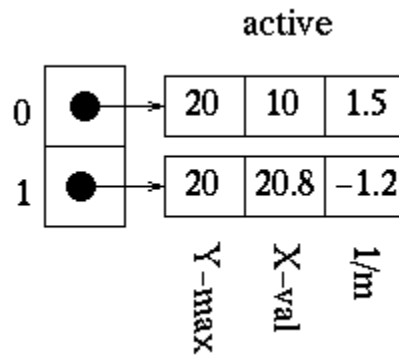
We then need to update the x values for all remaining edges.



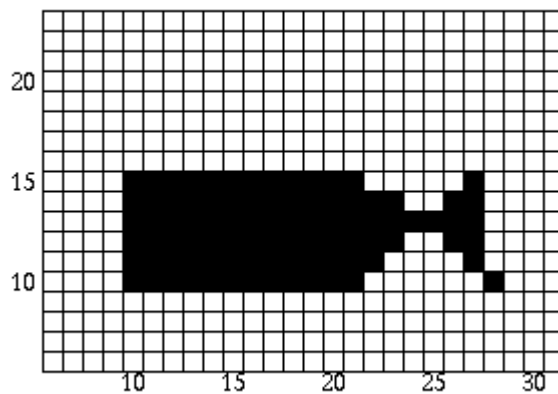
Now we can add the last edge from the global edge table to the active edge table since its minimum y value is equal to the next scan-line. The active edge table now look as follows (the global edge table is now empty):



These edges obviously need to be reordered. After reordering, the active edge table contains the following:

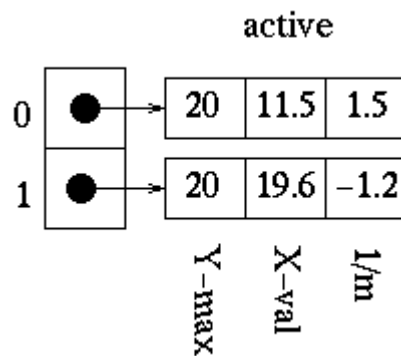


The polygon is now filled as follows:

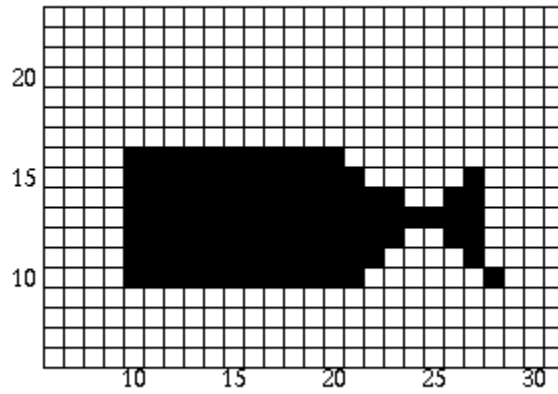


7. Scan-line = 16:

Once the first edge is encountered at  $x=10$ , parity = odd. All points are drawn from this point until the next edge is reached at  $x=21$ . We are now done with this scan-line. The  $x$  values are updated and the following is obtained:

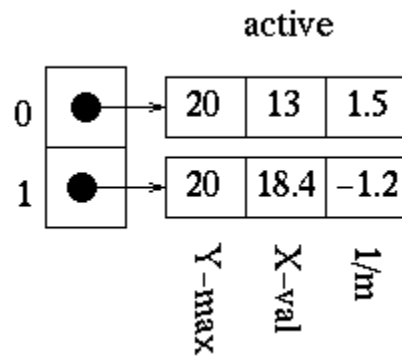


The polygon is now filled as follows:

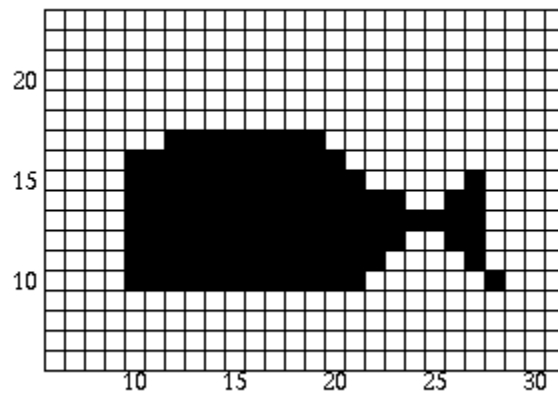


8. Scan-line = 17:

Once the first edge is encountered at  $x=12$ , parity = odd. All points are drawn from this point until the next edge is reached at  $x=20$ . We are now done with this scan-line. We update the  $x$  values and obtain:

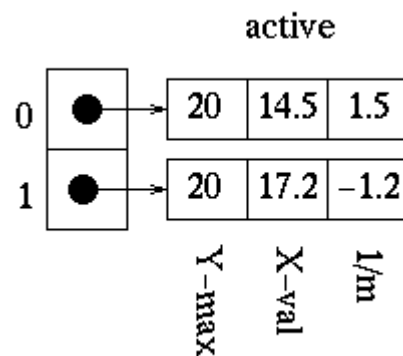


The polygon is now filled as follows:

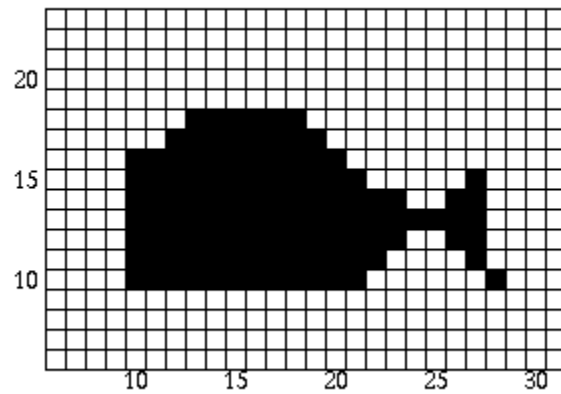


9. Scan-line = 18:

Once the first edge is encountered at  $x=13$ , parity = odd. All points are drawn from this point until the next edge is reached at  $x=19$ . We are now done with this scan-line. Upon updating the  $x$  values we get:



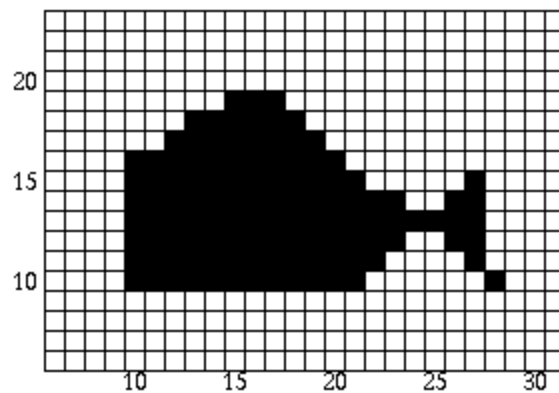
The polygon is now filled as follows:



10. Scan-line = 19:

Once the first edge is encountered at  $x=15$ , parity = odd. All points are drawn from this point until the next edge is reached at  $x=18$ . We are now done with this scan-line. Since the maximum  $y$  value for both edges in the active edge table is equal to the next scan-line, we remove them. The active edge table is now empty and we are now done.

The polygon is now filled as follows:



Now that we have filled the polygon, let's see what it looks like to the naked eye:



