Here I used :

1) TreeMap : I used tree map cause by using this data structure I can insert(),find() and other operation in O(log(n)) time. Where, in HashMap it would take O(n) time so TreeMap is clearly better choice. Below a comparison given :

	Hash Map	Tree Map
get()	O(n)	O(log(n))
put()	O(n)	O(log(n))
remove()	O(n)	O(log(n))

2)ArrayList : array list stores various kinds of objects(eg. points, vectors, edge etc). Here bellow a chart is given :

	Array List	Linked List
get()	O(1)	O(n)
add()	O(1)	O(1)
remove()	O(n)	O(n)

3)Priority Queue : And I used priority queue to sort data in custom way by implementing a comparator function .

4)Stack : Stack is used for triangulation. And also it takes O(1) time to push and pop operation so our total algorithm became O(n) time bounded.

2) I loop through the treetop and check whether sweep line has intersected any of edges from the TreeMap and also that edge is at the left from our current vertex. If any edge filled up those conditions then we looked for the immediate right vertices. If we got any vertex then we are done else we return null.

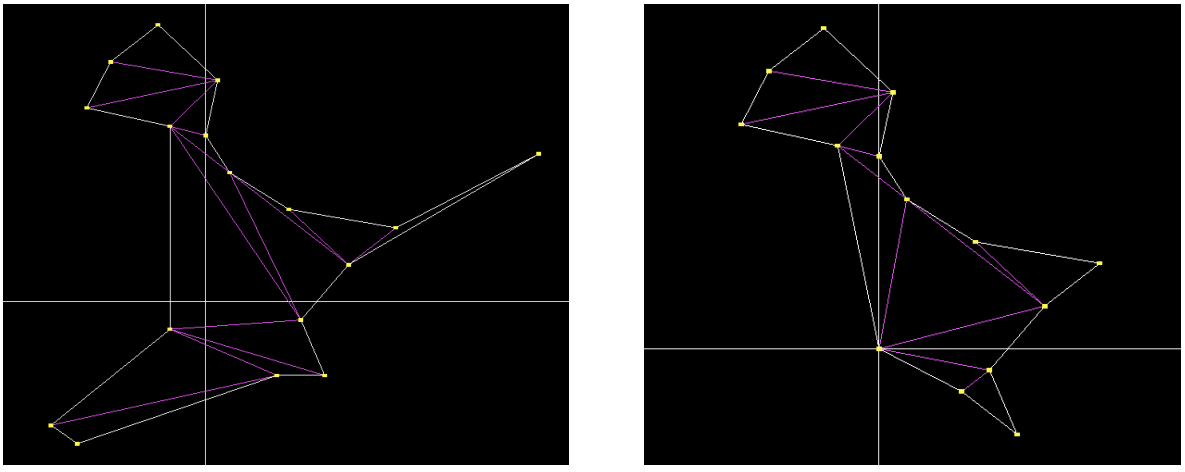
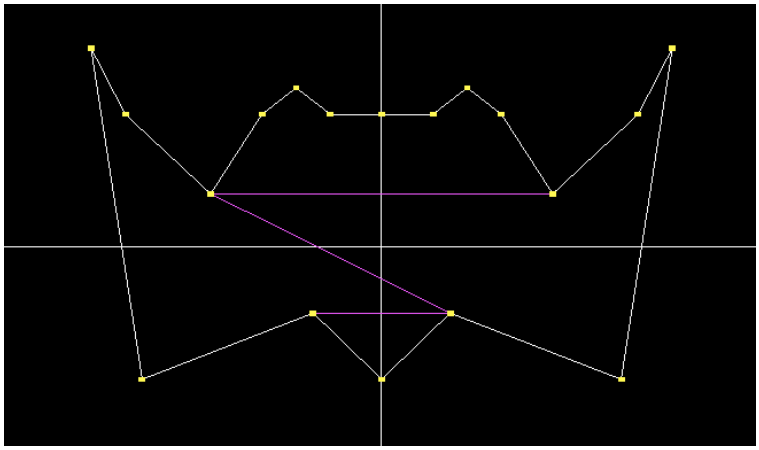
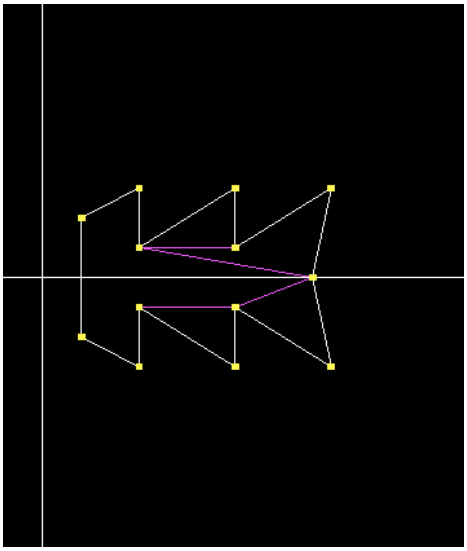


Figure 1: Sample Output for triangulating a y-monotone polygon.





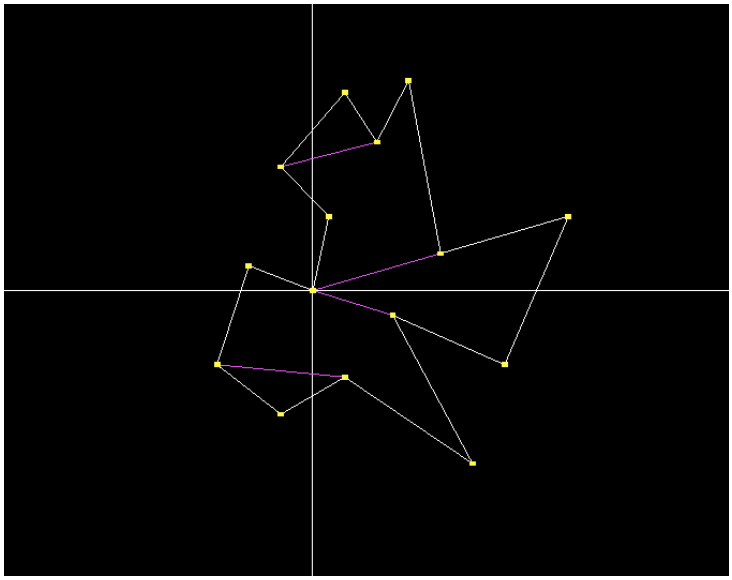


Figure 2: Sample Output for partitioning polygon in y-monotone pieces.