Homework 2 - Problem 1

Algorithm that finds the maximum possible number of lines crossed by a horizontal line through a shape.

```
Let Set A be an Set of the shape’s Y values
Let Set B be an Set of the indices of each value of A
Set index = 0
findMaxCrossing:
      Let Set C be a copy of A
      Use MERGESORT to sort C and B  //B sorted through C, Indexes of B align with C
      Set max, maxCurrent = 0;
      For n in C:
            Let point1 = B at same index as n
            Set rightPoint as the following point adjacent to n
            Set leftPoint as the point before adjacent to n

            If leftPoint has a different value to n then
                  If leftpoint is larger than n then  maxCurrent = maxCurrent + 1
                  Else maxCurrent = maxCurrent - 1;
            If rightPoint has a different value to n then
                  If rightpoint is larger than n then maxCurrent = maxCurrent + 1
                  Else maxCurrent = maxCurrent - 1;

            If n is not the same as the proceeding n and maxCurrent is larger than max then
                  max = maxCurrent
      Return max
```

This program produces the right output because it checks for all the points, and is sorted, meaning it will always check from one end to the other (smallest Y value -> largest Y value).

Time estimate: O(n Log n)
Proof (Master Theorem):

Using Case 2: $f(n)$ = Θ(n^(log a/ log b)),   2T(n/2) + n
Considering a: 2 , and b: 2
This means using Case 2 that $f(n)$ = Θ(n^(log 2/ log 2)) = n^1.