10/03/2024

Programming Assignment 1

CS 218 : Design and Analysis of Algorithms

Done by: NIKIL S

Algorithm

My algorithm is based on divide and conquer method similar to mergesort. So, I decided to name the two main functions that implement the algorithm as *merge()* and *mergesort()*. *mergesort()* function takes in parameters posters vector, starting index and ending index. It is a recursive function that merges two different outlines every time it calls *merge()*. *merge()* takes in parameters two vectors of Points, representing two outlines, and returns a single merged vector of Points. The *merge()* function is of **O(n)** while the *mergesort()* function is called **O(log(n))** times. So, effectively the overall order of the algorithm is **O(nlog(n))**.

The *merge()* function performs different operations to merge the two outlines by operating with two points at a time, each from the different outlines. I maintain a variable called *dom* that indicates whether the previously pushed point is of outline1 or outline2. The major two cases handled by it are:

- If the point to be pushed is of different outline from previous, then it pushes the intersection point and the point.
- Else, it pushes only the point.

The point to be pushed is determined by the y-value of both the outlines at that particular x-value, the higher value, is pushed.

Area and Length

The area and length required are calculated by the functions *area_of_outline()* and *lengthCovered_of_outline()* respectively.

Other functions

To help with the implementation of the algorithm, I have defined other functions like *solve_pt_on_line()*, *intersection_pt()*, and *remove_pts()*.