OSMAN MANTICI

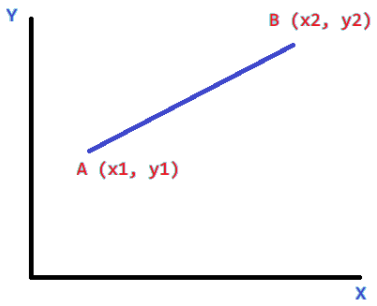
Algorithm Analysis Homework #1 Part 1 Report

This program takes an input as a file and creates an output again its file type. Input and output directories given by hand in the code.

After taking input start reading from given file line by line and arranges while reading circle objects(an object includes id, centerX, centerY, radius, and an arraylist) are created and setting informations about center coordinates and radius length. I use split() function in order to split space by space given input and then I set circle's variable field.

After input process is done, created circles' array given in to calculateOverlap() function as a paramater. This function calculates for every each circle whether overlaps or not with other circles, if there is an overlap occurs, the overlapped circles added their adjacency arraylist respectively.

In detail, calculateOverlap() function has a boolean flag variable. This variable set to true only one pair of circle overlaps. Also this variable sets from return value of distance() function. distance() function takes two circles and does a basic math calculation which is known as "distance of two points on the same plate".



$$AB = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

In order to the circles center coordinates(x,y points), the distance calculated given formula and its compared to addition of radiuses of circles. Let's their radiuses addition be "R" and the distance between them be "D" and therefore if and only if the circles are overlapped by the inequaty "D <= R", this means distance between centers must be equal or less than the radiuses addition. If so distance() function return true to the calculateOverlap() function. At the end the adjacency arraylist is created and arranged.

After all of these code come thorugh the bfs() function and this method takes two inputs. One of them first element of circle array which is hop distances are calculated between it and other circles, the other parameter is number of the circles.

At first line, the queue structure is structed and the parameter firstCircle added the queue. The levele set 0 because each level will be our hops. isVisited variable is also set true in order to not we will not visit it again. After that we visit firstly adjacent element of the taken circle and after

visiting all adjacencies we go through the new level and in other word hop level which is increased by 1. The queue works with first in first out manner and while being there an circle in the queue breadth first search will be continue.

The space complexity of this program is O(n+1) because there is no other space required. +1 is for the queue.

The time complexity of this program is O(n^2+V+E +L). n^2 is for calculateOverlap() method, V+E for the graph vertices and edges, L is the stand for the input line reading and this is real amount n+1 but input file reading may requires different time.