

Assignment-A Report

Pasan Jayawickrama 2016/CS/063

Introduction

Scenario A is used in advanced applications like 2D 3D games to setting display orders, visual surface detection in computer graphics, or order lines on their placement in the 2D plane.

In here after we give inputs to the program, it separates the line segments into two groups which are called front line segments and back line segments by inserting to a binary tree. To implement the program I used several data structures.

Programming Language

To implement this program I used Python 3.6.1 as the language. I choose Python because it makes implementation is easy and simple.

Used Data Structures

- Array
- Queue
- Binary Tree

Use of Array

In this program I use Array to hold line segments which are on line with our focusing line. Due to python does not support with array I choose python built in list feature to accomplish my task.

Use of Queue

A Queue is used in this scenario to hold input coordinates of 2D plane before insert them into the Binary tree. In here also used python built in list feature and data of the queue is list of tuples with coordinates. Ex: Queue $((AX, AY), (BX, BY)), ((CX, CY), (DX, DY)) \dots$

Binary Tree Structure

The best data structure and suitable for this scenario is binary tree. It is obvious there is two path front and back. Also choosing binary tree complexity of traveling through all nodes become $O(\log_2(n))$. So it makes efficient of searching.

Inputs to the Program

To give input to the program I make a text file called testCase.txt.

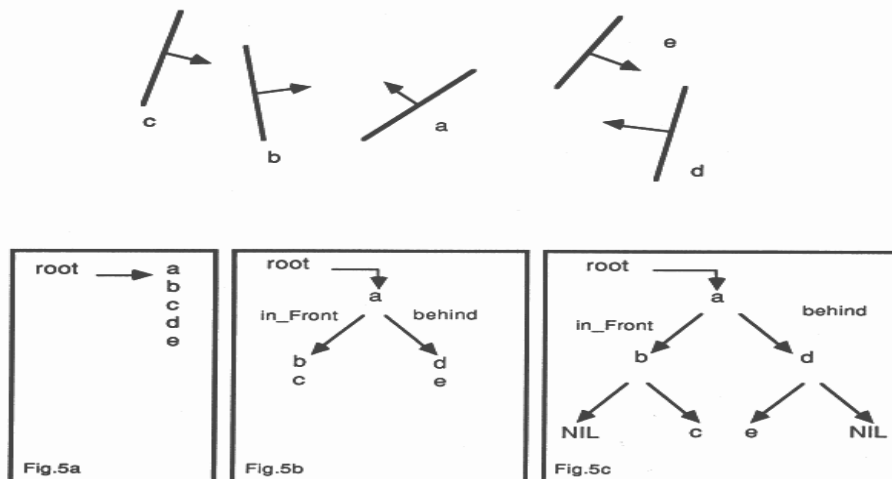
- First line of input is number of line segments in the 2D plane.
- Second line of input is the line number of starting line.
- From second line onwards contains coordinates of the line segments. Ex: (AX, AY), (BX, BY).

Output of the Program

- Front most line segment for given starting line.
- Print out line by line, from back to front line order in given scenario.
- Find front, back and on line segments for given line segment.

Methodology and the Process of the Program

First from the text file take number of line to iterate through input coordinates and insert into the queue, take starting line number and find corresponding coordinate and convert it to list of tuple with integer value and make coordinate root of the binary tree. Then comparing with root and popped out coordinate add to front or back of the root node by checking positions of lines. After all these comparing and inserting, by traveling through all nodes of binary tree take the output that needed.



Exceptions and Handled Way

When comparing lines there can found a line, that its part of line is in front and other part in back. So it is a special case that can found. In here I take intersect point of two lines and divide corresponding line into two parts and by comparing with current line insert them to front and back.

Another special case that can encounter is the line coplanar with comparing line. In this case coplanar line does not affect to our goal so that I put it in to the array of the comparing line node.