INTRO TO ALGORITHMS ASSIGNMENT 2 GROUP TASK

PEYTON GRATTINO & ALEX KUNZLER

Contents

1. Graphs

1a) What is the cost of the minimum spanning tree?

The cost of the minimum spanning tree is 19.

1b) How many minimum spanning trees does it have?

There are two ways you can achieve this by swapping the edge E-B for B-F.

1c) Tour of minimal spanning tree.

Node Start	Node End	Weight
A	E	1
E	F	1
B	F	2
B	E	2
F	G	3
G	H	3
C	G	4
B	C	5
C	F	5
D	G	5
A	B	6
C	D	6
D	H	7

They would be added in the following order

A-E
E-F
B-F
F-G
G-H
C-G
D-G

2a) Prim's Sort.

Node Start	Node End	Weight
A	B	1
B	C	2
C	D	3
A	E	4
E	F	5
F	G	1
G	H	1

2b) Kruskal's Algorithm.

Node Start	Node End	Weight
A	B	1
B	C	2
C	G	2
F	G	1
G	H	1
G	D	1
A	E	4

3.

If G's pointers > 1
Follow the pointers until they connect. If they connect, remove the most weighted edge connected to G.

2. Cost / Complexity / Big O

Equation A.
This is a divide and conquer algorithm, it has a running time of, **O(nLog(n))**

Equation B.
This is a recursive algorithm, and it has a run time of, **O(n)**

Equation C.
This is a 3-way merge sort, it has a running time of, **O(nLog(n))**

I would pick the divide and conquer, while it has a larger set-up that takes more memory it can sort the pieces quicker.

3. Algorithmic Efficiency

A linked sequence algorithm, you take the number you're looking for and find the spot in the array it should be, if it's not there then take the number you did find and go to the place in the array repeat until x is found. If you come back across a number you originally found, exit that loop and pick a different number, but the chance of this happening is very small.