Even Semester Term-II Examination, May-2021

Name - Gurjot Singh

Year - 2nd

Stream - B.Tech (CST)

Section - H

Class Roll Number - 23

Enrollment Number - 12019009022135

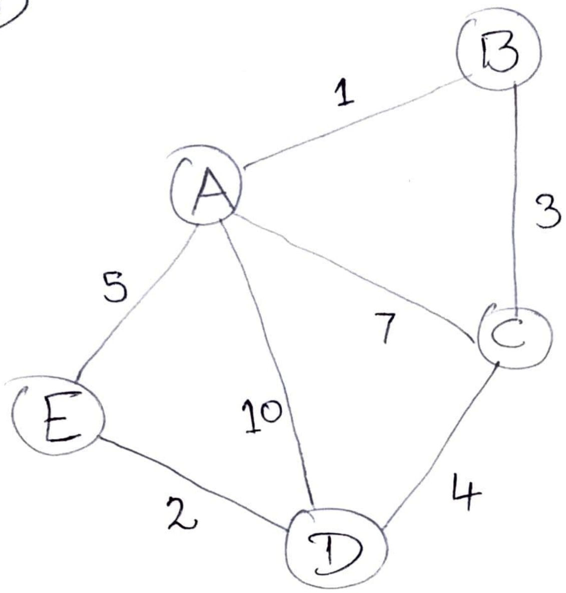
Paper Name - Design & Analysis Of Algorithm

Paper Code - PCCCS402

Signature - gurjot Singh

Date - 06/05/2021

1. B)

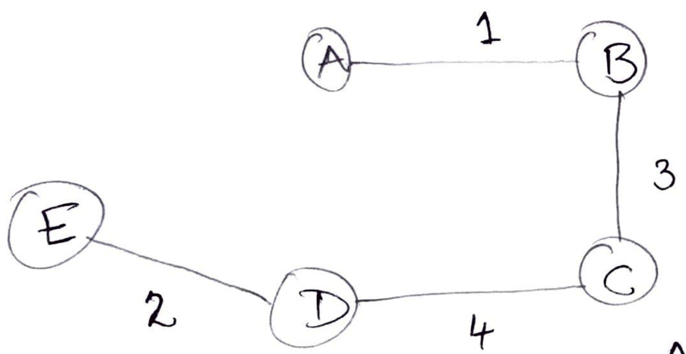



This is the graph.

No. of vertex = 5

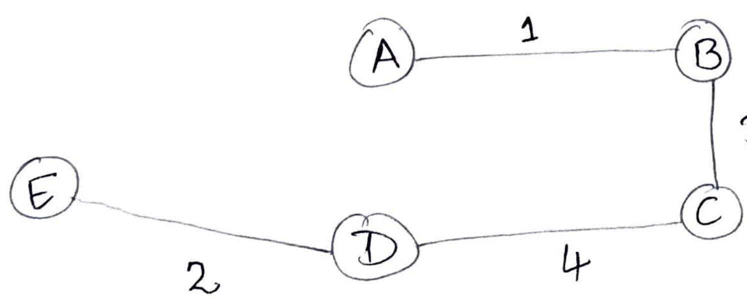
No. of edges = 4.

Min => A to B



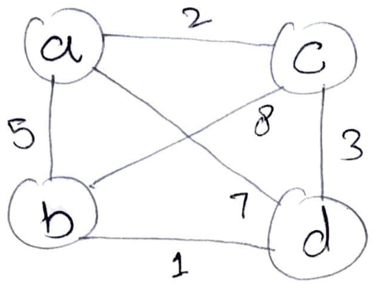

A → E, A → D, A → C not possible as it will form a cycle.

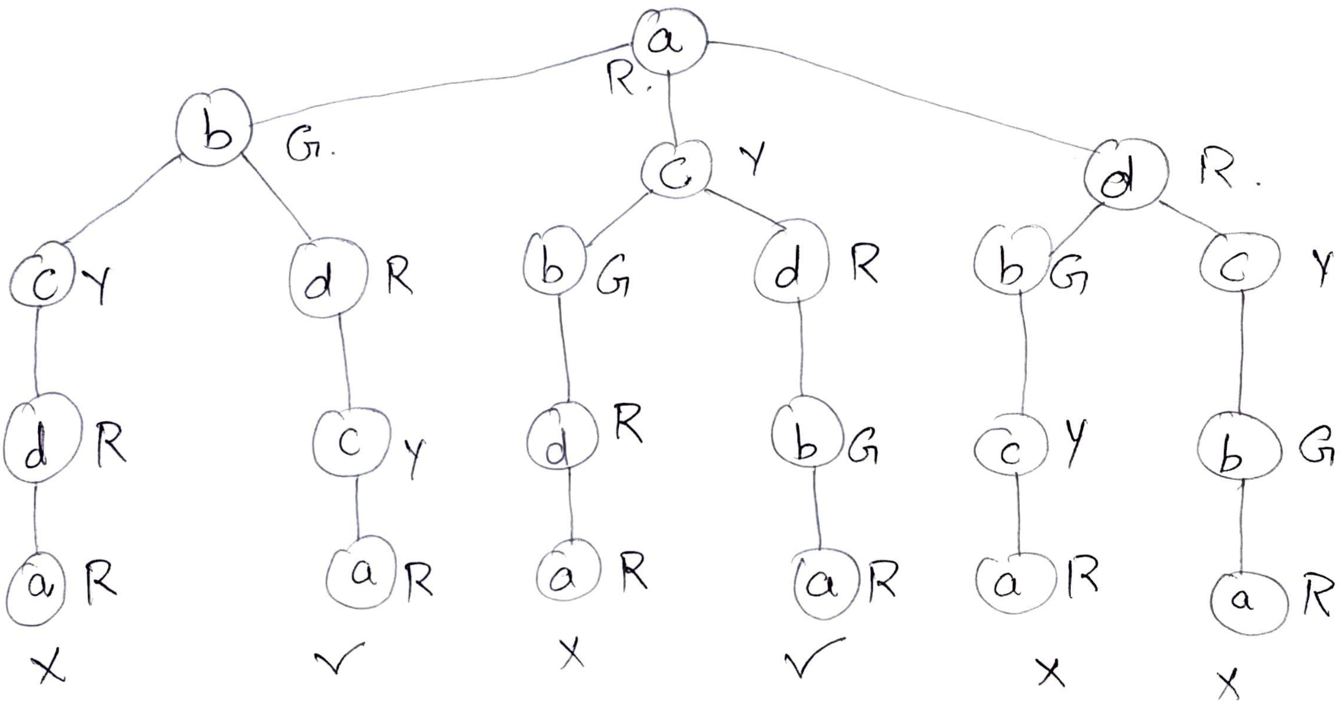
So, the minimum spinning tree is,




Total weight = 10.

2. A)



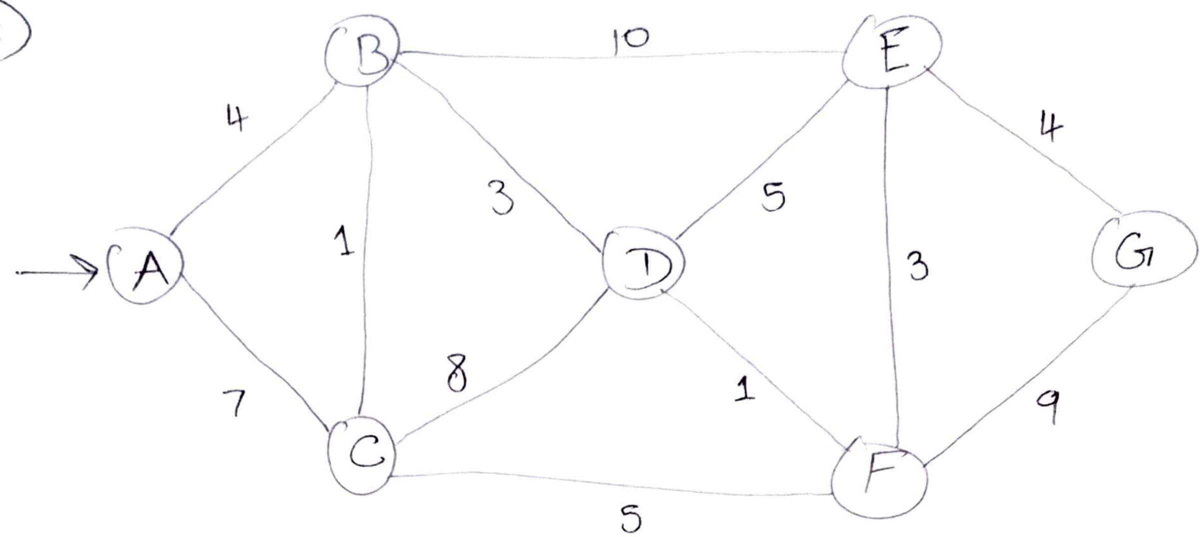



Hence the loop will be a – b – d – c – a

& a – c – d – b – a

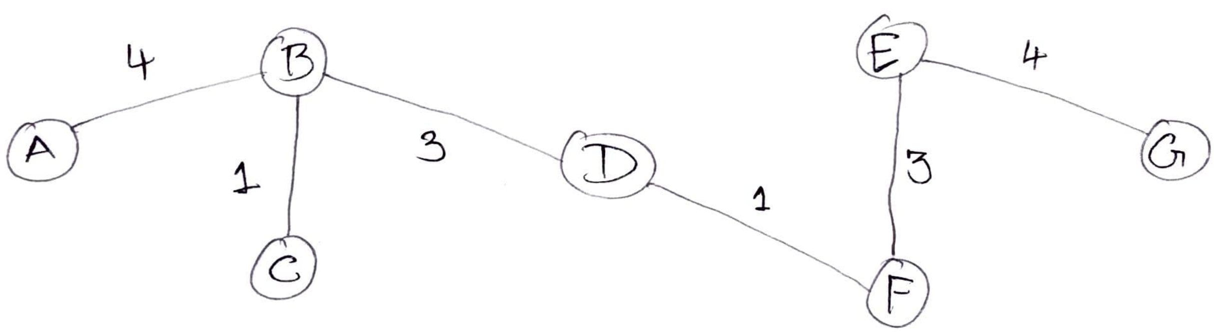
where a is Red, b is Green, c is yellow and d is Red.

3. A)




From A

Use Kruskal Algorithm. Find Minimum Spanning Tree From the given graph.



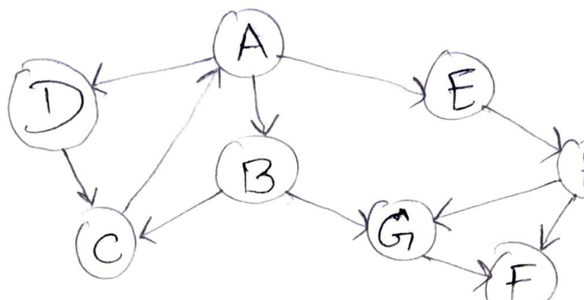

Minimum - $\longrightarrow$ Minimum SPANNING TREE

Travel Cost Is = Minimum Cost of Spanning Tree

Minimum Cost is = $(4+1+3+1+3+4)$

$= 16$ (ans)

4.b)




If there is ever a decision between multiple neighbour nodes in the BFS & DFS algorithm, we will always close the letter closest to the beginning of the alphabet first.

Process -

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

So, the Breadth First search for the given graph is

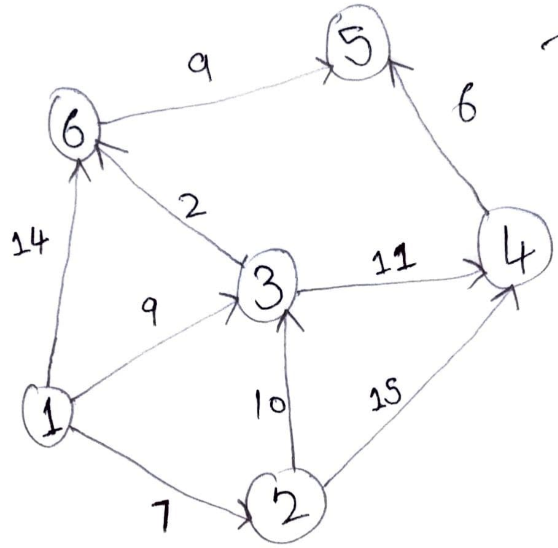
→ ABDCEGHF.

DFS is

→ ABCEHFGD.

~~So~~

5.B)

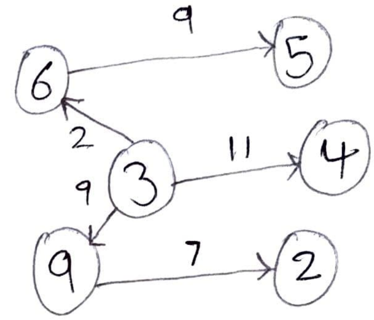



→ Dijkstra algo.

If $d(u) + c(u,v) < d(v)$

$d(v) = d(u) + c(u,v)$

Now representing :-




| Source | Destination 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|
| | ∞ | ∞ | ∞ | ∞ | ∞ |
| | 7 | 9 | ∞ | ∞ | 14 |
| 1, 2 | 7 | 9 | 22 | ∞ | 14 |
| 1, 2, 3 | 7 | 9 | 20 | ∞ | 11 |
| 1, 2, 3, 6 | 7 | 9 | 20 | ∞ | 11 |
| 1, 2, 3, 6 | 7 | 9 | 20 | ∞ | 11 |
| 1, 2, 3, 6, 4 | 7 | 9 | 20 | 20 | 11 |
| 1, 2, 3, 6, 4 | 7 | 9 | 20 | 20 | 11 |
| 1, 2, 3, 6, 4, 5 | 7 | 9 | 20 | 20 | 11 |

All Shortest Path from Source - 1.

6. A)

$I = (I_1, I_2, I_3, I_4, I_5)$

$W = (5, 10, 20, 30, 40)$

$V = (30, 20, 100, 90, 160)$

The Capacity of Knapsack $W = 60$

Now, fill the knapsack according to the decreasing value of $p_i$.

First we choose the item $I_1$, whose weight is 5

Then choose item $I_3$ whose weight is 20.

Now, the total weight of Knapsack is $20+5 = 25$

Now the next time is $I_5$ and its weight is 40 but we want only 35. So, we choose the fractional part of it,

i.e. $5 \times 5/5 + 20 \times 20/20 + 40 \times 35/40$

weight $= 5 + 20 + 35 = 60$

Maximum Values :-

$30 \times 5/5 + 100 \times 20/20 + 160 \times 35/40$

$= 30 + 100 + 140 = 270$ (Minimum Cost)

| ITEM | $w_i$ | $v_i$ |
|---|---|---|
| $I_1$ | 5 | 30 |
| $I_2$ | 10 | 20 |
| $I_3$ | 20 | 100 |
| $I_4$ | 30 | 90 |
| $I_5$ | 40 | 160 |

Taking value per weight ratio i.e. $P_i = v_i/w_i$

| ITEM | $W_i$ | $V_i$ | $P_i = V_i/W_i$ |
|---|---|---|---|
| $I_1$ | 5 | 30 | 6.0 |
| $I_2$ | 10 | 20 | 2.0 |
| $I_3$ | 20 | 100 | 5.0 |
| $I_4$ | 30 | 90 | 3.0 |
| $I_5$ | 40 | 160 | 4.0 |

Now, arrange the value of $P_i$ in decreasing order.

| ITEM | $W_i$ | $V_i$ | $P_i = V_i/W_i$ |
|---|---|---|---|
| $I_1$ | 5 | 30 | 6.0 |
| $I_3$ | 20 | 100 | 5.0 |
| $I_5$ | 40 | 160 | 4.0 |
| $I_4$ | 30 | 90 | 3.0 |
| $I_2$ | 10 | 20 | 2.0 |

7. B) **Tractable Problem** - A problem that is solved by a polynomial-time algorithm.
The upper bound is polynomial.

**For Example** - •) Searching an unsorted list.
•) Searching an ordered list.
•) Sorting a list.
•) Multiplication of integers.
•) Finding minimum spanning tree in a graph.

**Intractable Problem** - A problem that cannot be solved by a polynomial-time algorithm. The lower bound is exponential.

From a computational complexity stance, intractable problems are problems for which there exists no efficient algorithm to solve them.

**For Example** - •) Towers of Hanoi, we can prove that any algorithm that solves this problem must have a worst-case running time that is atleast $2^n - 1$.
•) List of all permutations of n numbers.