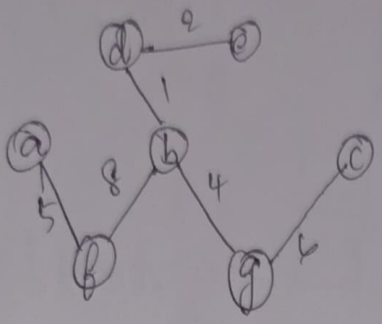
Kruskal's algorithm:

Started with least pair



MST=26

Time complexity $O(V)$

Prim's algorithm → faster

Compared to Kruskal

Huffman code: frequency analyzing of each and every character & to find bit code and to use encode & decode.

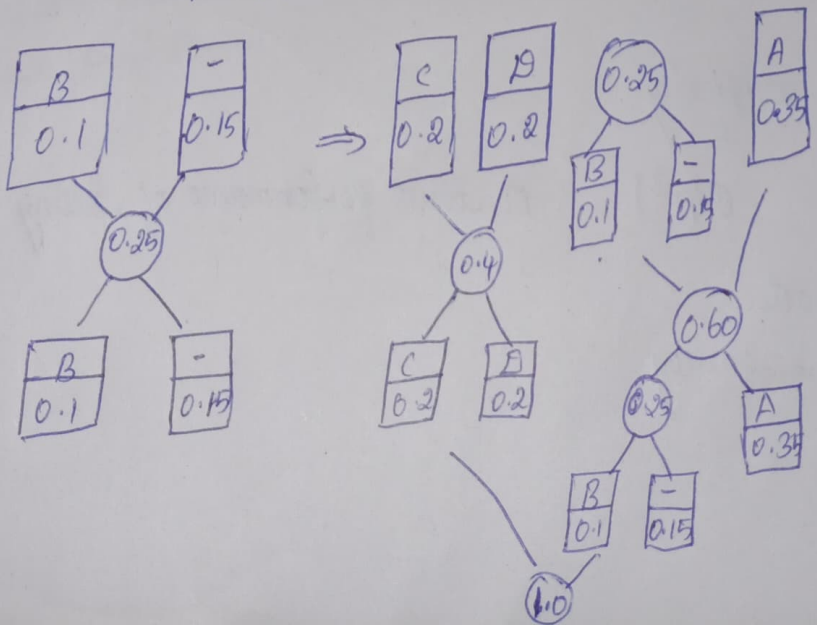
Huffman Trees: → it is used for data compression → it is efficient techniques it is used in cryptography

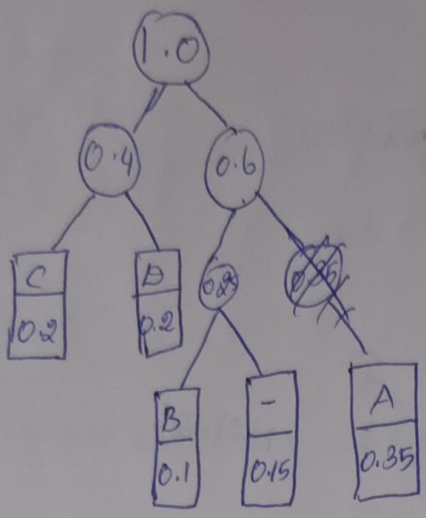
Character	A	B	C	D	-
probability	0.35	0.1	0.2	0.2	0.15

Probability → chance of occurance

Encode - DAD

Decode - 100|0|1|0|1|1|0|1





left - 0
right - 1

Encode
↓
Convert the text into cyber text.

Encode - DAD

	A	B	C	D	-
	0.35	0.1	0.2	0.2	0.15
Code	11	100	00	01	101

DAD → 01 11 01

Decode

100 11 01 11 01

B A D - A D

Time complexity:

$O(n^3)$ n times performance of sorting

Characte
frequency