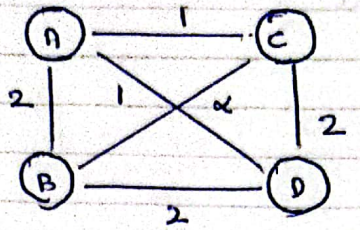
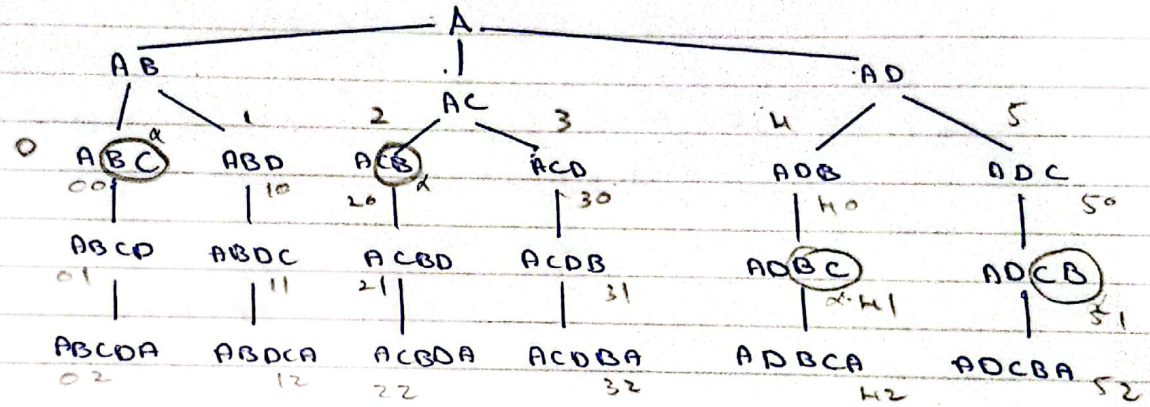
A-3.

1.



Decision Tree.



~~Rough Hypothesis~~:

~~def cost_to_go(city, history[3]):~~

~~if history[city] not in history:~~

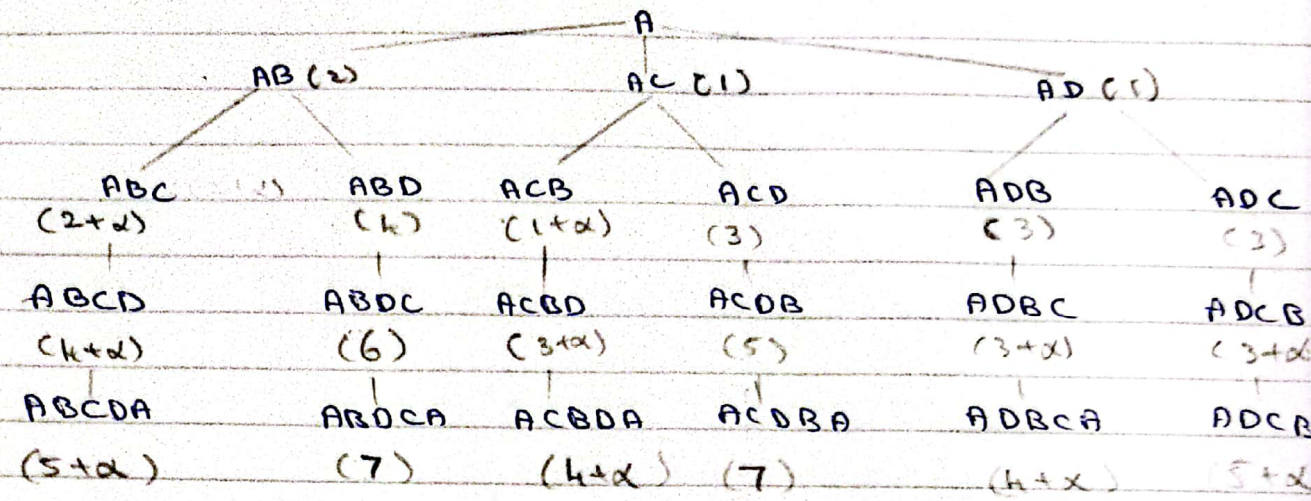
~~history[city] = cost to go~~

	0 A	1 B	2 C	3 D
0 A	0	2	1	~~∞~~ 1
1 B	2	0	~~∞~~	2
2 C	1	~~∞~~	0	∞
3 D	~~∞~~ 1	2	∞	0

=> Cost of visiting city in - City (two cities).

From the code.

a)



b) if $\alpha \sim [1, 2]$

The paths ACBDA & ADBCA will be optimal since the cost-to-go for both the paths will be 6.

2. a) i) recover Path.

input: s, g, pred.

$s \in V$, $g \in V$

Pred is a dictionary.

output: the sequence of vertices on the optimal path

let it be a list of element $\in V$.