CS & IT ENGINEERING

Algorithms

Dynamic Programming (DP)

DPP 01 (Discussion Notes)

By- Aditya sir

[MCQ]

#Q. What is the time complexity of dynamic programming for matrix chain multiplication problem?

A $O(n^2)$ ✗

B $O(n^3)$ ✓

C $O(n\log n)$ ✗

D None of these ✗

Ans : B

Matrix Chain Multiplication

MCM.

$$(A_1 A_2 \cdots A_n)$$

$n \rightarrow$ no. of matrices

$\Rightarrow O(n * n^2)$

$\Rightarrow \boxed{O(n^3)}$

DP based MCM

$\longrightarrow O(n^3)$

Space Complexity $\longrightarrow O(n^2)$

[NAT]

#Q. Consider the matrices x, y and z with dimension 10 × 20, 20 × 30 and 30 × 40 respectively. Then what is the minimum number of multiplications required to multiply the matrices? _______

10x20 20x30 30x40

Matrix Chain Multiplication (MCM)

(XYZ)

① ((XY)Z)

18000

② (X(YZ))

32000

Ans :- 18000

① $(XY)Z$

$X \rightarrow 10 \times 20$

$Y \rightarrow 20 \times 30$

$Z \rightarrow 30 \times 40$

$$A_{m \times n} * B_{n \times p} = C_{m \times p}$$

no. of mul $= \underline{\underline{m \times n \times p}}$

$\Rightarrow$

(XY)

$\underbrace{10 \times \underline{20} \quad 20 \times \underline{30}}$

$\underbrace{10 \times 30 \quad 30 \times 40}$

$$= 10 \times 20 \times 30 + 10 \times 30 \times 40$$

$$= 6000 + 12000$$

$$= \underline{\underline{18000}}$$

2) $X(42)$

10×20 $\quad$ 20×30 $\quad$ 30×40

20×40

$20 \times 30 \times 40$

$+$

$10 \times 20 \times 40$

$= 24000$

$+$

8000

$= \underline{\underline{32000}}$

[MCQ]

#Q. What is the length of the LCS for the pair of strings given below.

P = ATGACTATAA

Q = GACTAATA

A) 5

B) 6

C) 7

D) 8

Ans: C

Longest Common Subsequence

(LCS)

Approaches:

1) manually

2) Top-Down

3) Bottom-up

Idea :-

$A[1 \ldots n]$ $\qquad$ $B[1 \ldots m]$

i $\qquad$ j

$$LCS(i, j) = 1 + LCS(i-1, j-1), \quad \underline{\underline{A[i] = B[j]}}$$

$$LCS(i, j) = \max \begin{Bmatrix} LCS(i-1, j), \\ LCS(i, j-1) \end{Bmatrix}, \quad A[i] \neq B[j]$$

Soln: Bottom-up: Tabulation.

P = ATGACTATAA

Q = GACTAATA

			A	T	G	A	C	T	A	T	A	A
		0	1	2	3	4	5	6	7	8	9	10
	0	0	0	0	0	0	0	0	0	0	0	0
G	1	0	0	0	1	1	1	1	1	1	1	1
A	2	0	1	1	1	2	2	2	2	2	2	2
C	3	0	1	1	1	2	3	3	3	3	3	3
T	4	0	1	2	2	2	3	4	4	4	4	4
A	5	0	1	2	2	3	3	4	5	5	5	5
A	6	0	1	2	2	3	3	4	5	5	6	6
T	7	0	1	2	3	3	3	4	5	6	6	6
A	8	0	1	2	3	4	4	4	5	6	7	7

Ans

LCS: GACTATA $\Rightarrow$ 7

$$\begin{tikzcd}
L[i-1,j-1] & L[i-1,j] \\
L[i,j-1] & L[i,j] \arrow[l] \arrow[u] \arrow[ul]
\end{tikzcd}$$

match : $1 + L[i-1, j-1]$ (diagonal)

not match : $\max\{\text{Left}, \text{Up}\}$

Appr 2 : Manually :

P = ATGACTATAA → n = 10

Q = GACTAATA → m = 8

1 2 3 – – . . 10

min (m, n) = min (10, 8) = 8

8 7 6 5 4 3 2 1 0

GACTATA

[MCQ]

#Q. Consider a connected weighted graph G = (V, E), where |V| = n, |E| = m, if all the edges have distinct positive integer weights, then the maximum number of minimum weight spanning trees in the graph is ?

A n

B m

C 1

D n^{n-2}

Ans: C

n → vertices

m → edges.

Distinct pos wts

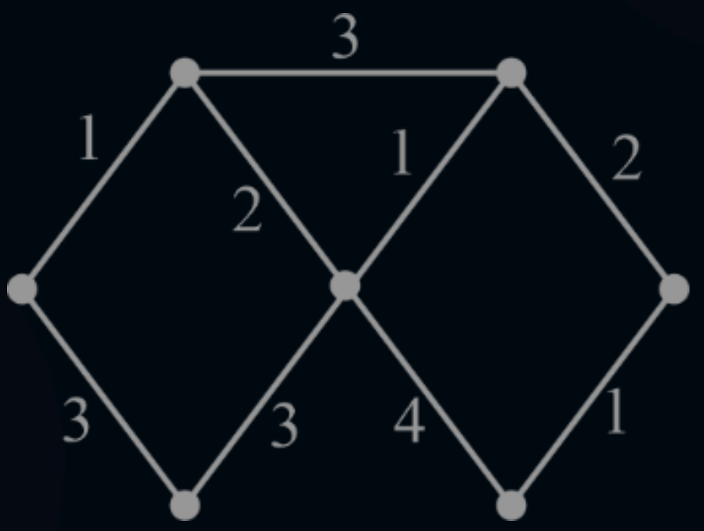
↳ Prims
↳ Kruskal

Structure & Min Cost

↳ both same.

[NAT]

#Q. What is the weight of the minimum spanning tree for the graph shown below?



Ans: 10

Born

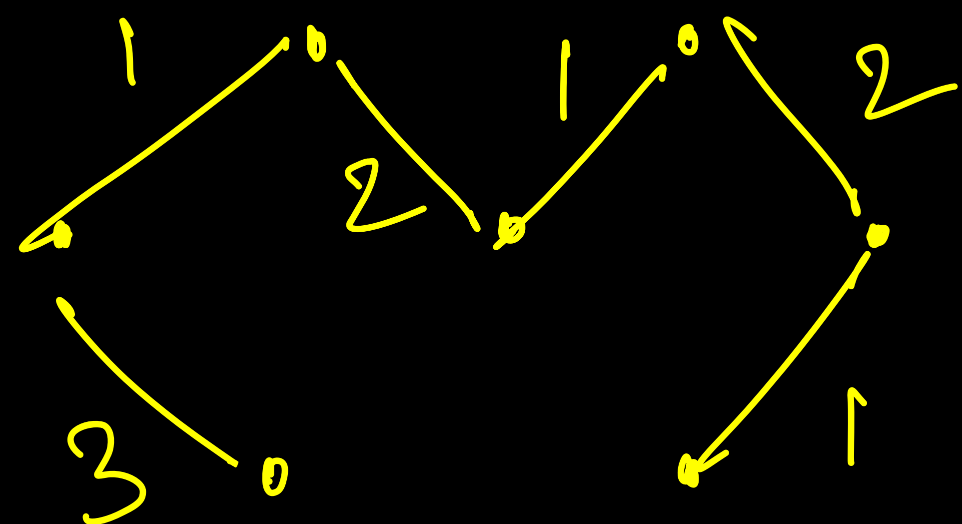
Dijkstra MCST

MCST cost = 1+1+1+2+2+3

= $\boxed{10}$

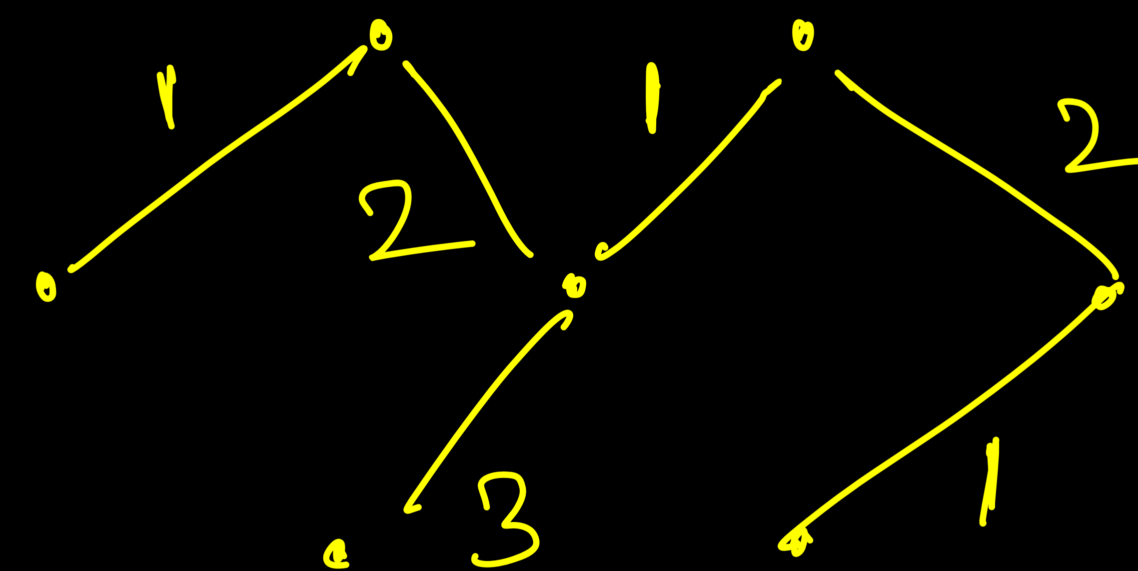
Soln:-

Kruskal:



$$\text{MCST Cost} = \underline{1+1+1} + \underline{2+2} + 3$$
$$= 3+4+3$$
$$= \boxed{10}$$

Prims



$$\text{MCST cost} = 1+1+1+2+2+3$$
$$= \boxed{10}$$

[MCQ]

#Q. How many minimum spanning tree does this graph have?

A 2

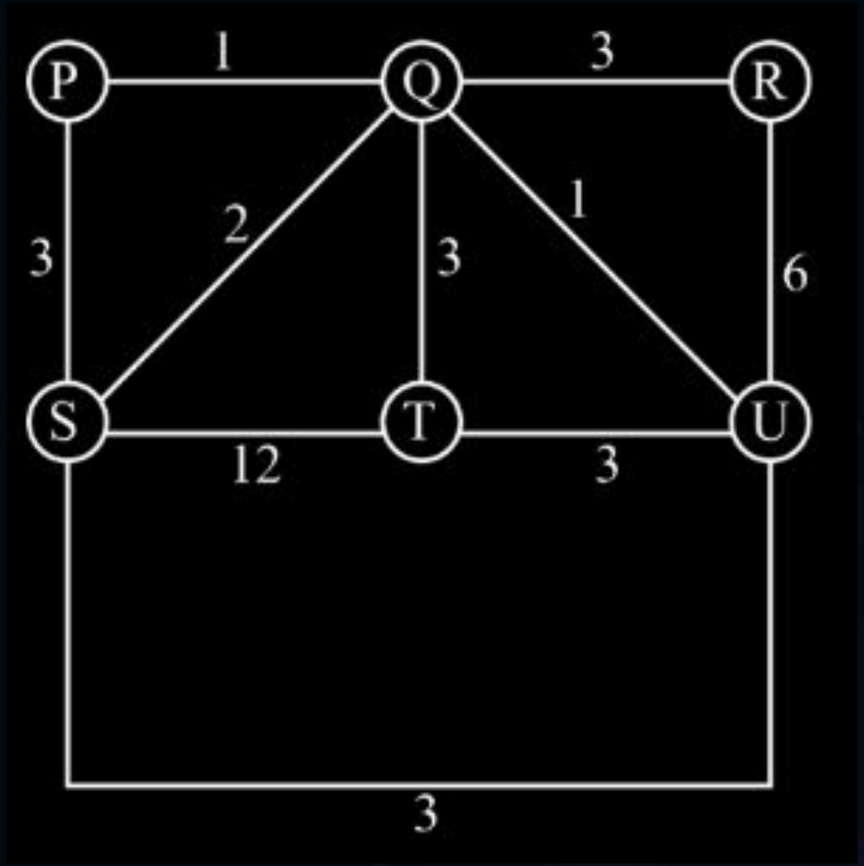
B 3 ✗

C 4 ✗

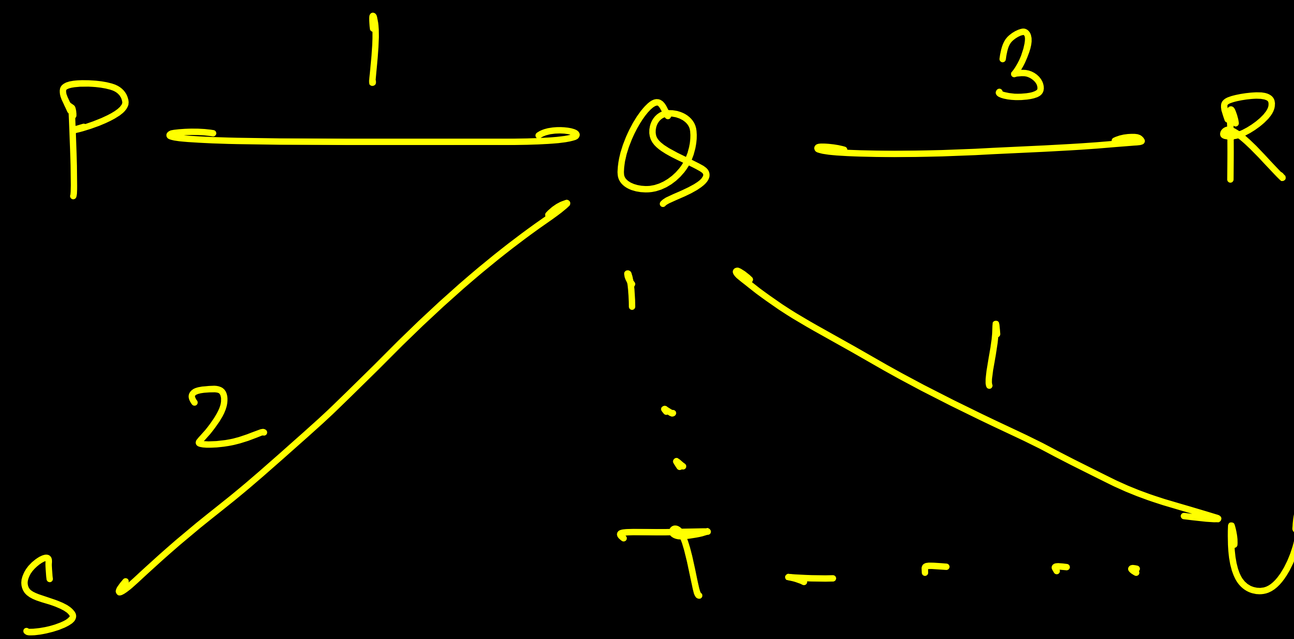
D 5 ✗

2 MCSTs

Ans: A



Soln:

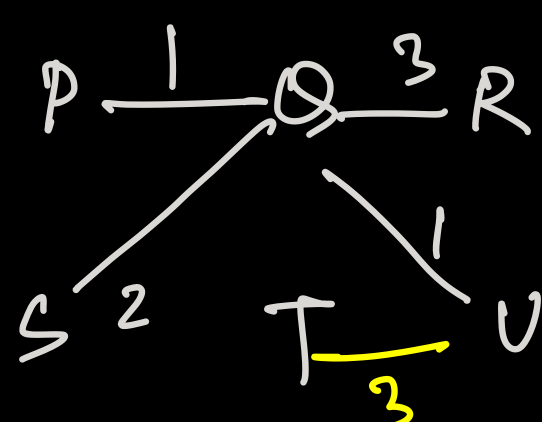


n = 6

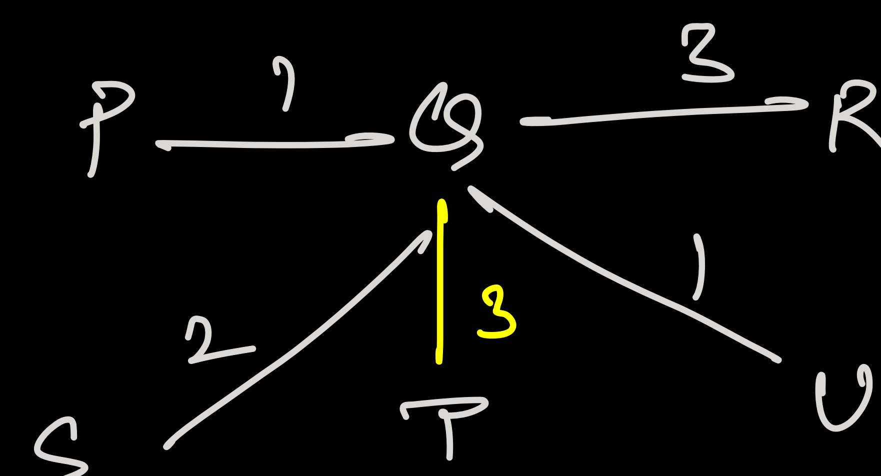
edges in MCST = 6 - 1

$= \boxed{5}$

$^2C_1$



OR



2 MCSTs

[MCQ]

#Q. Consider the following problem with knapsack capacity of 8

Item	Profits	Weights
I_1	13	1
I_2	8	5
I_3	7	3
I_4	3	4

$n = 4$

$M = 8$

Which of the following item is not selected in the optimal solution of 0/1, knapsack problem?

C I_1 only

C I_2 only

D I_3 only

D I_4 only

Ans: C

0/1 → Binary Knapsack → Tabulation.

n → (n+1) x (M+1)
M

Recurrence :-

$$knap[i, j] = knap(i-1, j) \quad , \; w[i] > M$$

$$= \max \left\{ \underline{knap(i-1}, j) \; , \; knap(i-1, j - w[i]) + P[i] \right\}$$

when $w[i] \le M$

$\theta \rightarrow m$

$\omega \rightarrow n$

obj	P	W	n
			0
1	13	1 ✓	1
2	8	5	2
3	7	3 ✓	3
4	3	4 ✓	4

m

n \ m	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	
1	0	13	13	13	13	13	13	13	13	
2	0	13	13	13	13	13	21	21	21	
3	0	13	13	13	20	20	21	21	21	
4	0	13	13	13	20	20	21	21	23	

max

Ans:

$\boxed{I_1,\ I_3,\ I_4}$ → optimal Profit

[MCQ]

#Q. Consider the following statements

S1: for every weighted graph and any two vertices p and q, Bellman ford algorithm starting at p will always return a shortest path to q. → not always (eg when -ve wt cycle is present)

False

S2: Dijkstra greedy algorithm for single source shortest path can be used to solve the all pairs shortest path problem. → True.

True

Which of the statement is correct?

C only S_1 ✗

C only S_2 ✓

D Both S_1 and S_2 are true ✗

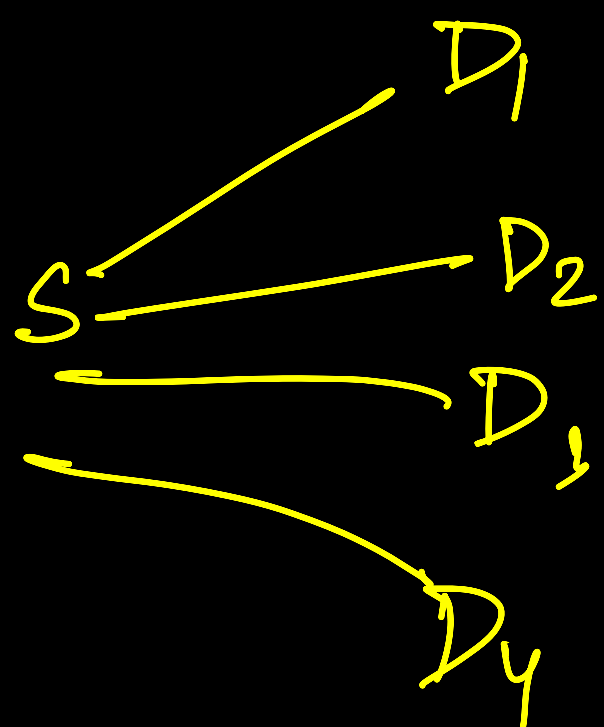
D neither S_1 nor S_2 is true ✗

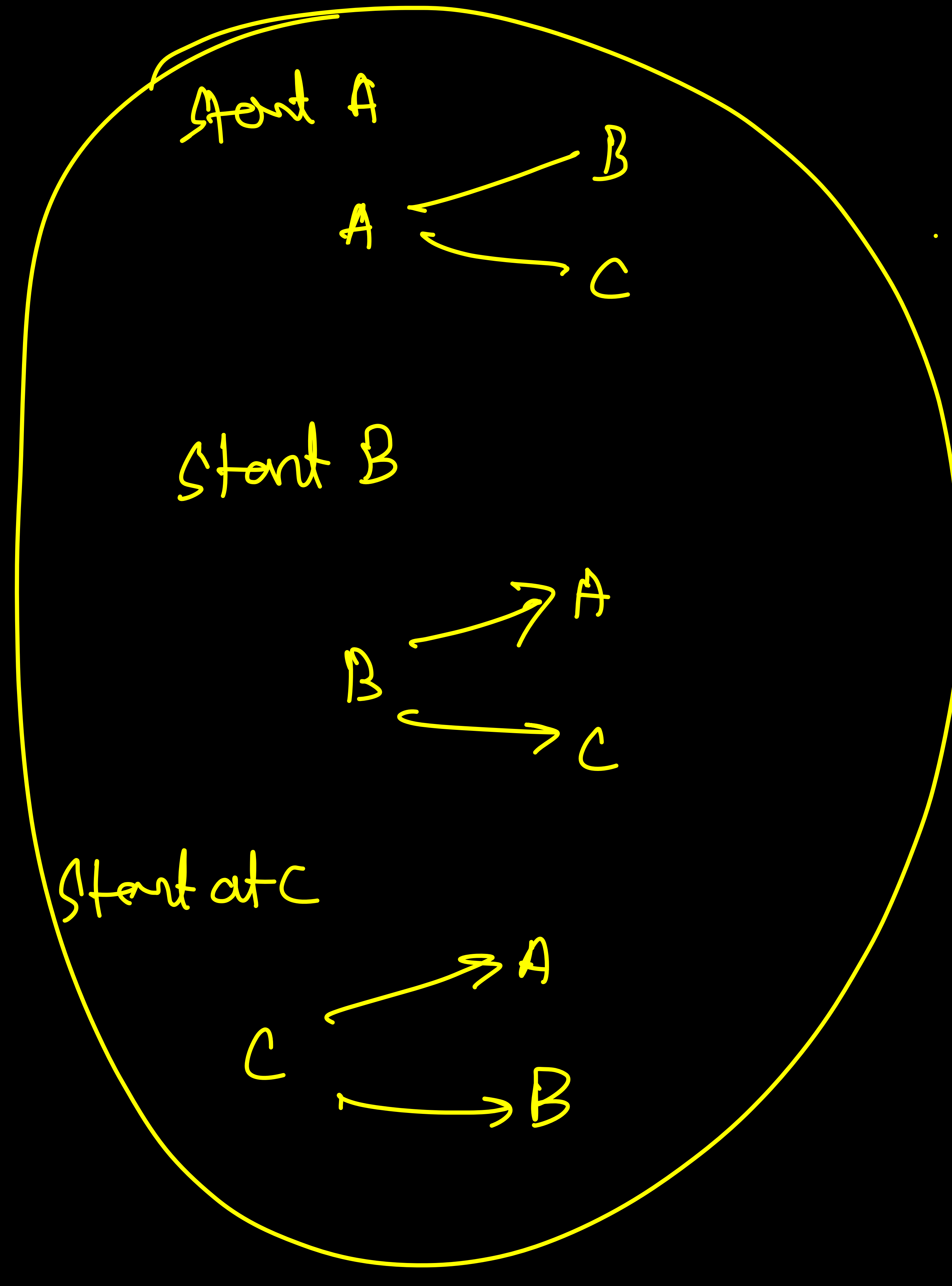
Ans: C

Single Source Shortest Path :-

		+ve wt edges	-ve wt edges But no -ve wt cycle	-ve wt cycle
1) Dijkstra	$\rightarrow$ Greedy	✓	✗	✗
2) Bellman Ford	$\rightarrow$ DP	✓	✓	✗

SSSP D_1





APSP

THANK - YOU