CS & IT ENGINEERING

Algorithms

Greedy Method

Lecture No.- 06

By- Dr. Khaleel Khan sir

Recap of Previous Lecture

Topic

- Shortest Paths
- Dijkstras Algorithm

Topics to be Covered

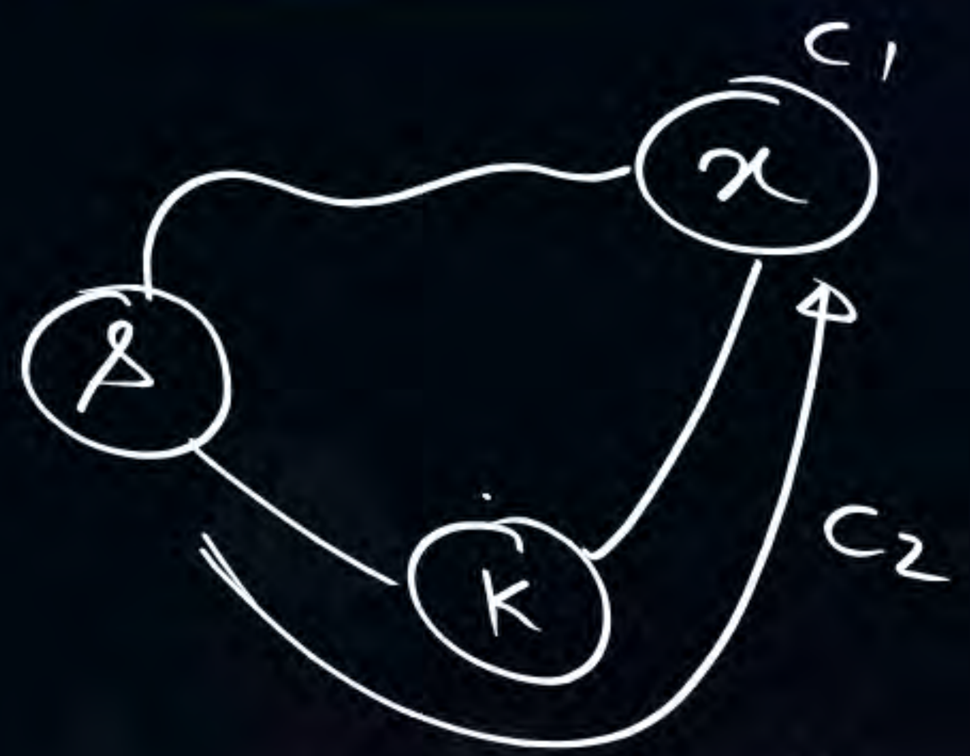
Topic Single Source Shortest Paths

Problem Solving

Topic : Greedy Method

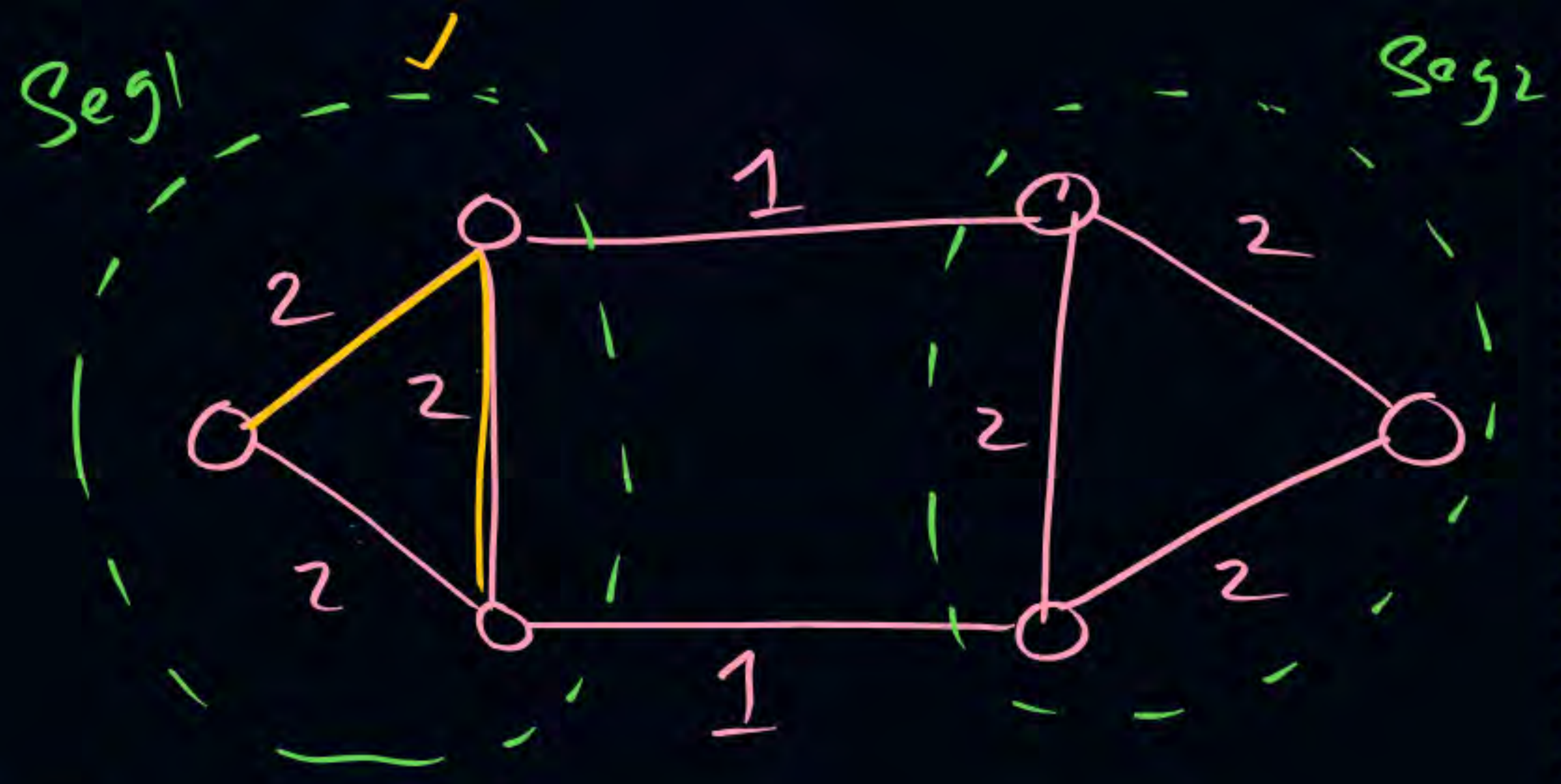
vertex sel	d-values 1	2	3	- - .	n

(i) Relaxation Process

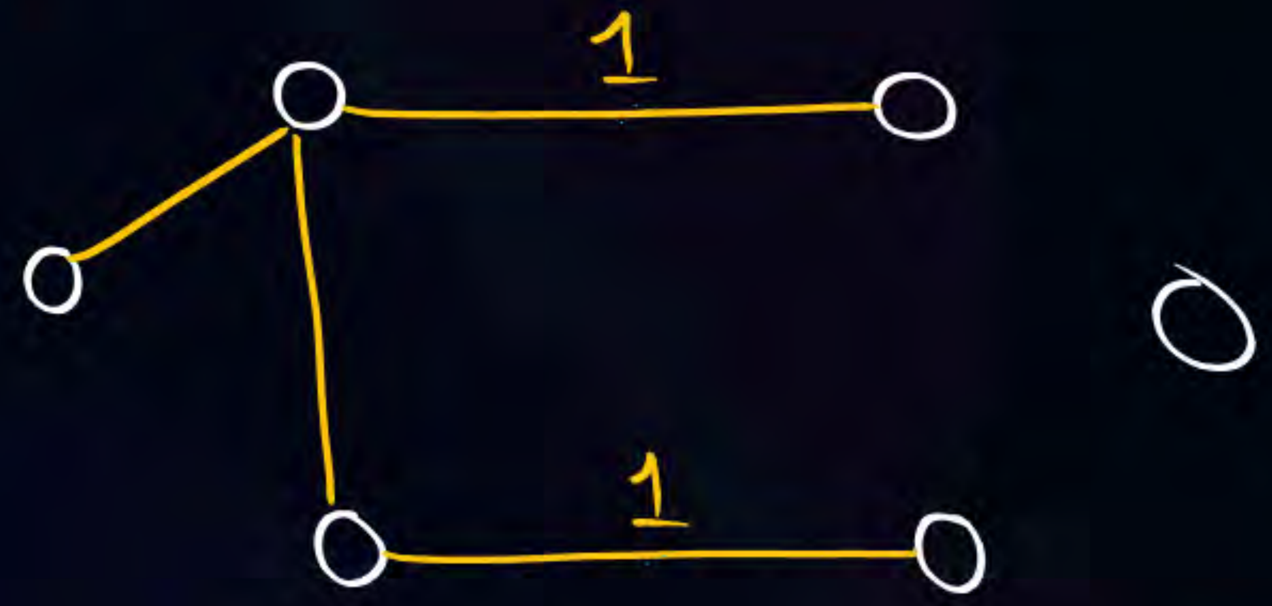


$if\ (c_2 < c_1)$

$d[x] = c_2$



Combinatorics



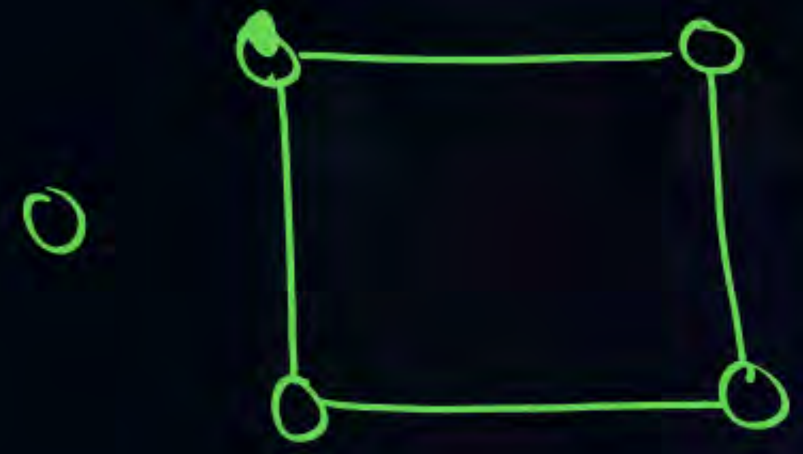
$$\left[{}^{2}C_1 * {}^{3}C_2 * {}^{2}C_1\right] = 12$$

Ans: 12

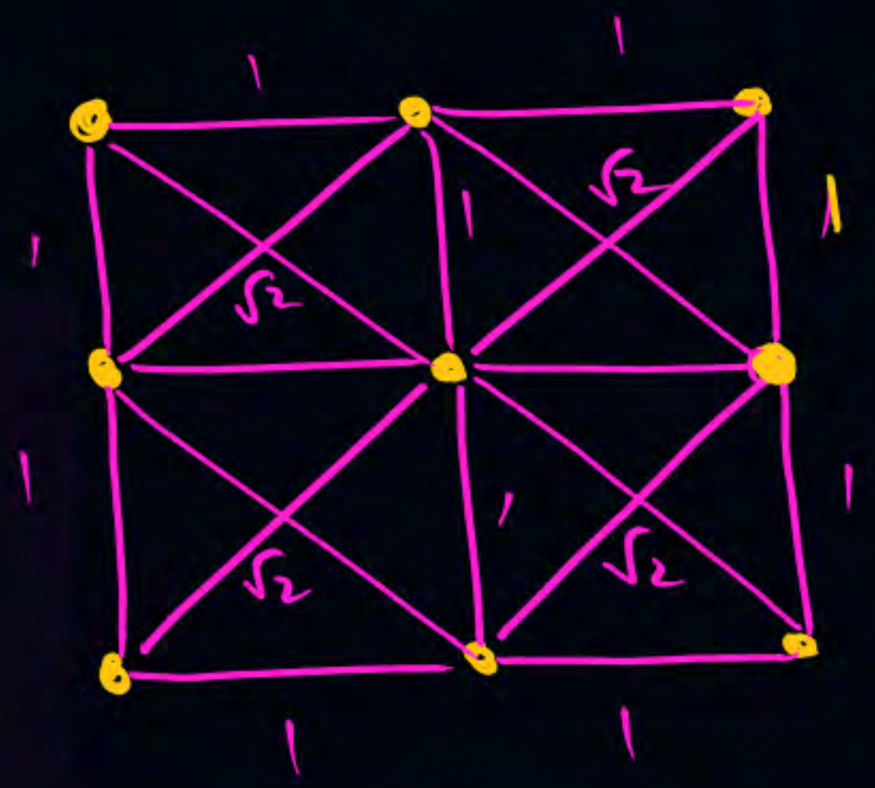
$$\left[{}^{6}C_3 - 8\right] = 12$$

$$\boxed{{}^{6}C_3 - \left(\text{All combinations leading to cycle}\right.}$$

${}^{4}C_1$



No. of M. Cost Sp. Trees



$n = 3$

Topic : Greedy Method

H/W

Let G(V, E) be a directed graph, where V = {1, 2, 3, 4, 5} is the set of vertices and E is the set of directed edges, as defined by the following adjacency matrix A.

Lower Triangular Matrix

$n=3$ $n=4$

$$A[i][j] = \begin{cases} 1, & 1 \leq j \leq i \leq 5 \\ 0, & otherwise \end{cases}$$

A[i][j] = 1 indicates a directed edge from node i to node j. A directed spanning tree of G, rooted at r ∈ V, is defined as a subgraph T of G such that the undirected version of T is a tree, and T contains a directed path from r to every other vertex in V. The number of such directed spanning trees rooted at vertex 5 is ______.

[GATE-2022: 2M]

Topic : Greedy Method

Heap: Time : $O(n^2)$

src → [ ] → d-values

```
Algorithm ShortestPaths (v, cost, dist, n)
// dist [j] , 1 ≤ j ≤ n, is set to the length of the shortest
// path from vertex v to vertex j in a digraph G with n
// vertices. dist[v] is set to zero.- G is represented by its
// cost adjacency matrix cost/.[1 : n, 1 : n].
{
1.  for i := 1 to n do
    { // Initialize S.
        S[i] := false; dist[i] := cost[v , i];
    }
2. S[v] := true; dist[v] := 0.0; // Put v in S.
3. for num := 2 to n — 1 do
   {
      // Determine n — 1 paths from v.
   a) Choose u from among those vertices not
      in S such that dist[u] is minimum;
   b) S[u] := true; // Put u in S.
   c) for (each w adjacent to u with S[w] = false) do {
        // Update distances.
        if (dist [w] > dist[u] + cost[u, w])) then
        dist[w]:= dist[u] + cost[u, w];
      }
}
```

$S[1..n] = F$

Lines 7–10: $O(n)$

Line 11: $O(1)$

Line 12: (n) : $O(n^2)$

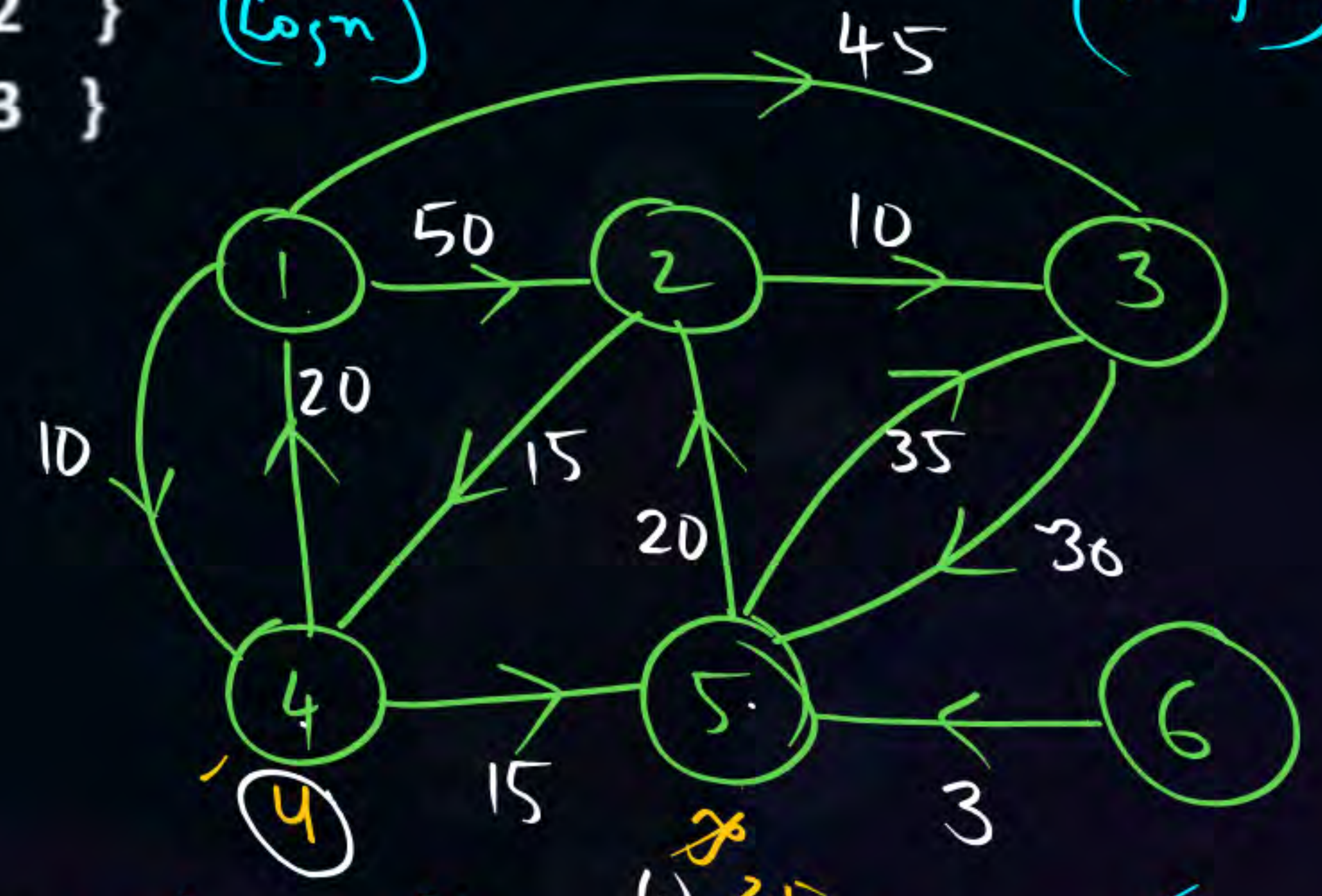
Line 15: (n) Heap: $\log n$

Line 17: (1)

Line 18: n n (n)

Lines 19–21: $(\log n)$ $(n^2 \log n)$

$v = 1$



Time: $n + n \cdot \log n + n^2 \log n = (n+e) \cdot \log n$

Topic : Greedy Method

Consider a weighted undirected graph with positive edge weights and let uv be an edge in the graph. It is known that the shortest path from the source vertex s to u has weight 53 and the shortest path from s to v has weight 65. Which one of the following statements is always true?

(A) weight (u, v) < 12 (B) weight (u, v) ≤ 12

(C) weight (u, v) > 12

(D) weight (u, v) ≥ 12

$\boxed{wt[u,v] \geq 12}$

12

Topic : III. Greedy Method

a) SACEGT ✓b) SACET
c) SDT d) SBDT

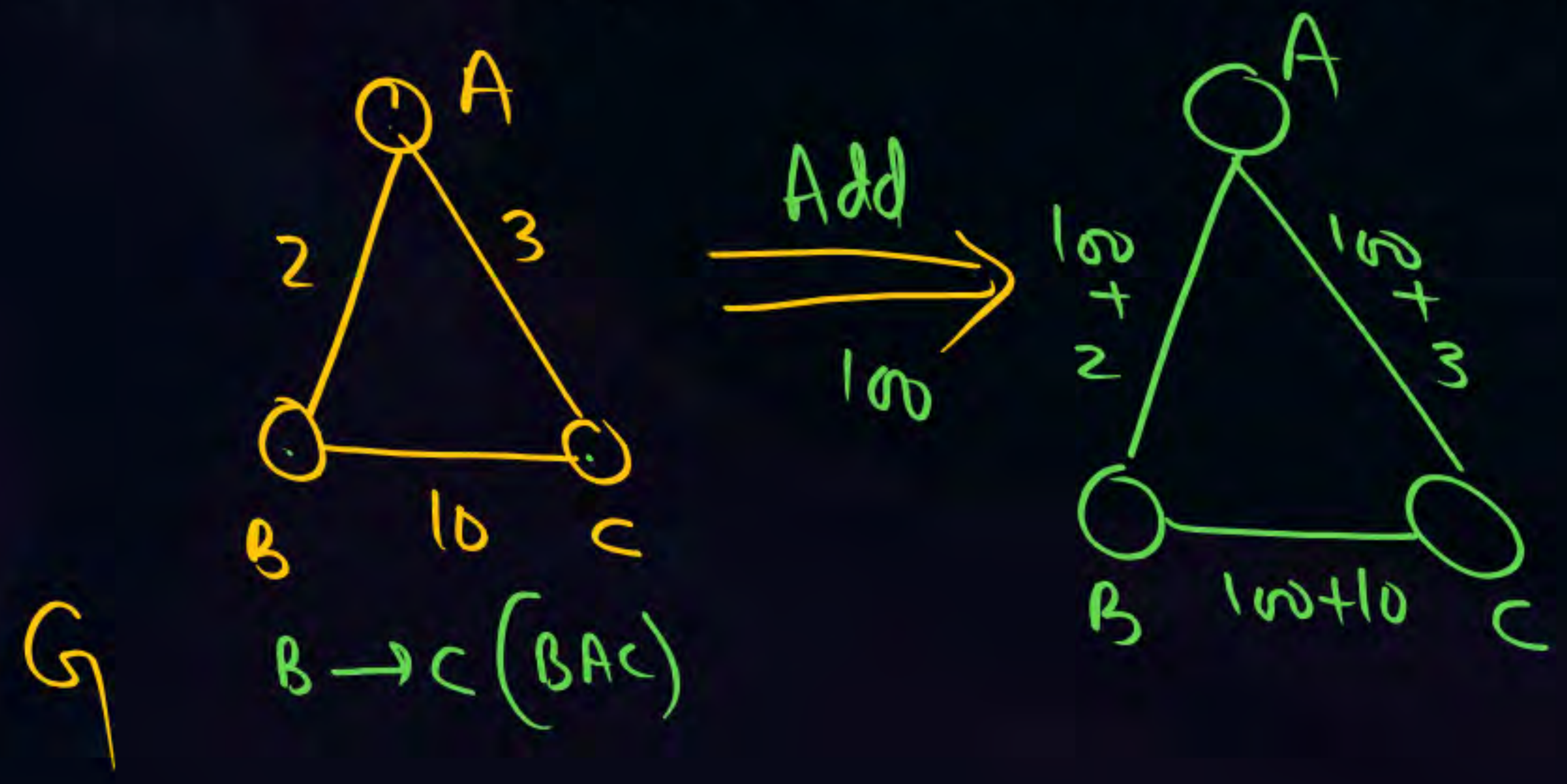
Q. Applying Dijkstra's Algorithm over the given Graph, Which path is reported from 'S' to 'T';

* (2M)

⟨SACET⟩ = 10

Q. Let G be a weighted connected undirected graph with distinct positive edge weights. If every edge weight is increased by the same value, then which of the following statements is/are true?

1. Minimum spanning Tree of the graph does not change. : T
2. Shortest path between any pair of vertices does not change. : F

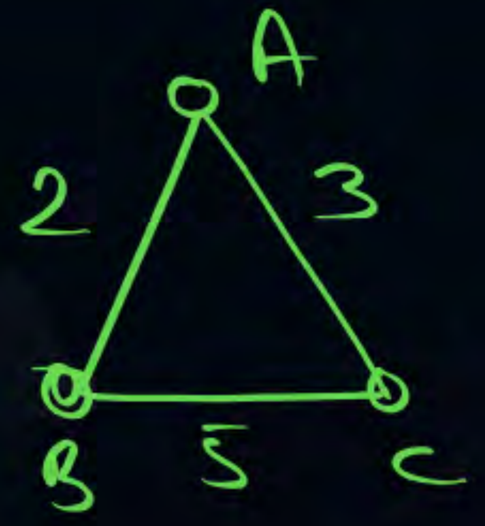


Topic : III. Greedy Method

Q. Let G = (V, E) be any connected undirected edge-weighted graph. The weights of the edges in E are positive and distinct. Consider the following statements:

(I) Minimum Spanning Tree of G is always unique. T

(II) Shortest path between any two vertices of G is always unique. F



Which of the above statements is/are necessarily true?

(a) (I) only

(b) (II) only

(c) Both (I) and (II)

(d) Neither (I) nor (II)

Topic : Greedy Method

1. Consider the following table:

Algorithms	Design Paradigms
(P) Kruskal	(i) Divide and Conquer
(Q) Quick sort	(ii) Greedy
(R) Floyd-Warshall	(iii) Dynamic Programming

Match the algorithms to the design paradigms they are based on.

(a) (P) $\leftrightarrow$ (ii), (Q) $\leftrightarrow$ (iii), (R) $\leftrightarrow$ (i)

(b) (P) $\leftrightarrow$ (iii), (Q) $\leftrightarrow$ (i), (R) $\leftrightarrow$ (ii)

(c) (P) $\leftrightarrow$ (ii), (Q) $\leftrightarrow$ (i), (R) $\leftrightarrow$ (iii)

(d) (P) $\leftrightarrow$ (i), (Q) $\leftrightarrow$ (ii), (R) $\leftrightarrow$ (iii)

Dynamic Programming (DP)

: design Strategy proposed by
Richard Bellman

<Rand Corporation>

NOT Coding

(Tabulating the values of Subproblems)

Etymology:

(Planning over a Period of Time)

Topic : Algorithms: DP

Dynamic Programming (DP) is an algorithm design strategy (method) paradigm, used for solving program, whose solution are viewed as a result of marking a sequence of decisions.

Terminology:

- Problem Definition
- Constraints
  - Implicit (Local) → Feasibility criteria (Correct (Impl. constraints))
  - Explicit : (Boundary)
- Solution space (∞, 2^n, n!.....)
- Feasible solution :
- Objective function : min / max → A given criteria
- Optimal solution: Unique (value)

<Problems>

- <Decision> T/F → (Feasible Soln)
- <Optimization> ↓ (Value) KNAP

Topic : Dynamic Programming: (DP)

Dynamic programming (DP) is an algorithm design method used for solving problems, whose solutions are viewed as a result of making a set/sequence of decisions;

- One way of making these decisions is to make them one at a time in a step-wise (sequential) step-by-step manner and never make an erroneous decision. This is true of all problems solvable by Greedy method.
- For many other problems it is not possible to make step-wise decisions based on local information available at every step, In such a manner that the sequence of decision made is optimal.

① COIN CHANGE PROBLEM:

Given a set of Coin values; (lens)
Construct a sum of Money using as few coins as possible. We can use each coin value any number of times;

Coins : $\langle c_1, c_2, c_3 \ldots c_k \rangle$

Target Money : N

Ex 1: (N = 12); Coins = {1, 2, 5}

✓ Greedy Method : 5 + 5 + 2 = 12

Ex: 2: N = 6 ; Coins = {1, 3, 4}

Greedy method: 4 + 1 + 1 = 6

GM failed Optimal Soln : 3 + 3 = 6

Consider the weights and values of items listed below. Note that there is only one unit of each item.

$M = 11$

0/1 KNAPSACK :

Using Greedy Method (P/W)

Item number	Weight (in kgs)	Value (in Rupees)
1	10	60
2	7	28
3	4	20
4	2	24

$\times\ \frac{P_1}{W_1} = \frac{60}{10} = 6$

$\times\ \frac{P_2}{W_2} = \frac{28}{7} = 4$

$\checkmark\ \frac{P_3}{W_3} = \frac{20}{4} = 5$

$\checkmark\ \frac{P_4}{W_4} = \frac{24}{2} = 12$

$2 + 4$

$x_4 = 1$

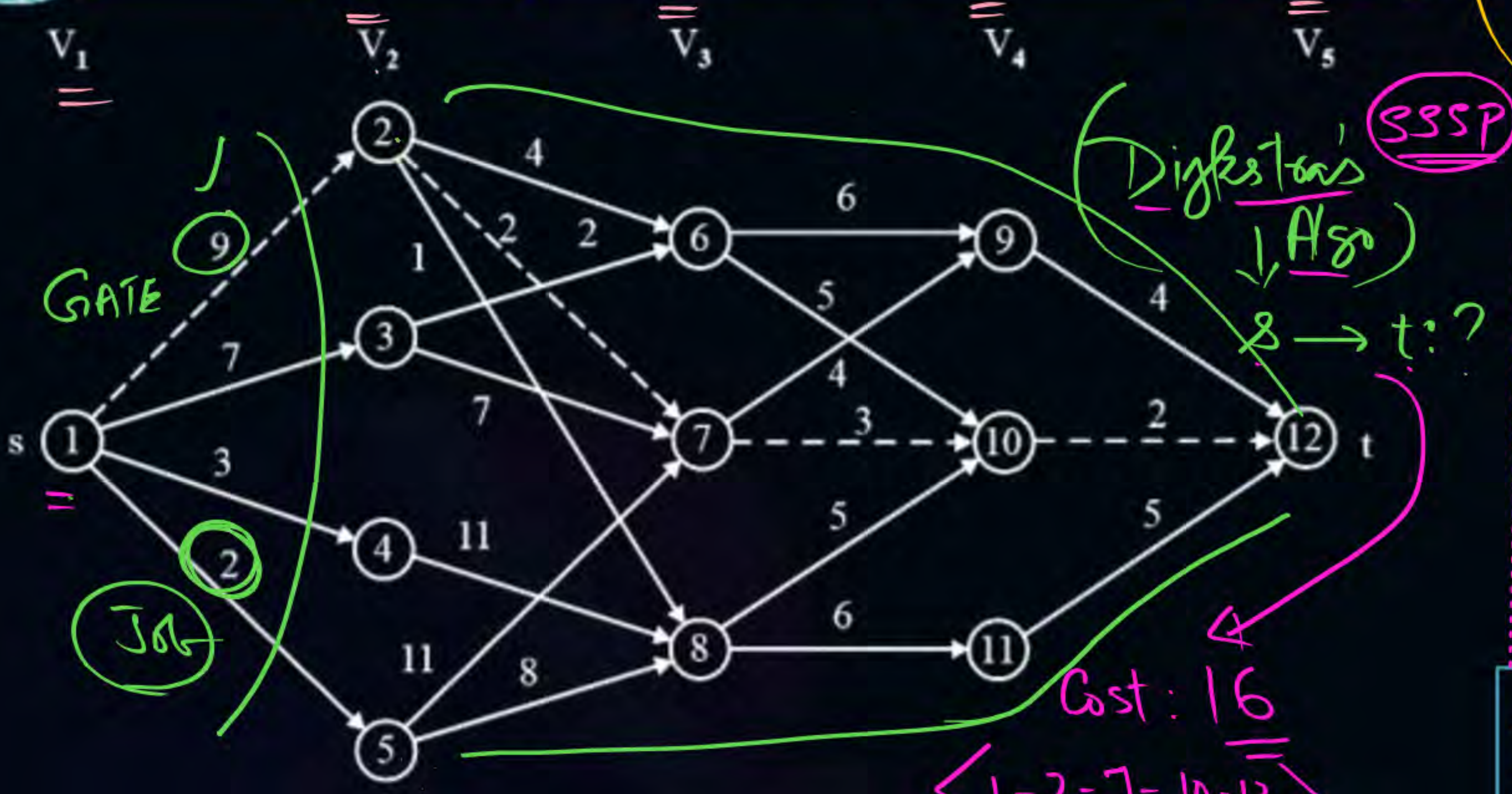
$x_3 = 1$

44

$V_{opt} : 60$

(Multi - Stage Graph)

S.P.S.P



$V_i \rightarrow V_{i+1}$

Finding Shortest Path from 's' to 't'

Cost: 16

⟨1-2-7-10-12⟩

⟨1-3-6-10-12⟩

G = (V, E)

Greedy Method:

$\boxed{1-5-8-10-12}$ = 17

1-2-7-10-12 : 16 ✓

GM Fails

THANK - YOU