18/10/22

GOODLUCK Page No. Date

★ Single source shortest Paths

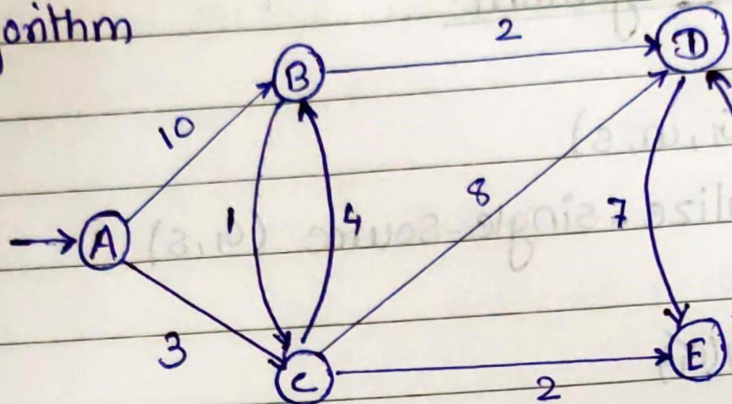
1) Dijkstra's algorithm

Dijkstra (G, ω, s)
    initialize-single-source (G, s) — ①
    $S = \phi$
    $Q = V(G)$
    while ($Q \neq \phi$)
        u = Extract-min(Q)
        $S = S \cup \{u\}$
        for each vertex $v \in adj(u)$
            RELAX (u, v, ω) — ②

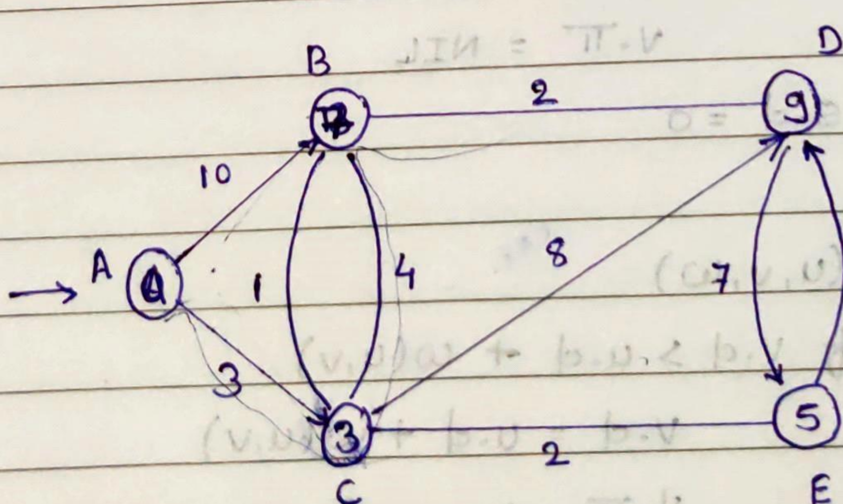
1) Initialize-single-source (G, s)
    for each vertex $v \in V(G)$
        $v.d = \infty$
        $v.\pi = NIL$
    $s.d = 0$

2) RELAX (u, v, ω)
    if $v.d > u.d + \omega(u,v)$
        $v.d = u.d + \omega(u,v)$
        $v.\pi = u$

1. Find shortest path from source vertex A to all the vertices of directed graph using dijkstra's algorithm



Q →	A	B	C	D	E
	$\boxed{0}$	∞	∞	∞	∞
	-	10	$\boxed{3}$	∞	∞
	-	7	-	11	$\boxed{5}$
	-	$\boxed{7}$	-	11	-
	-	-	-	$\boxed{9}$	-
	-	-	-	-	-



Shortest paths

$d(A,B) = d(A,C) + d(C,B)$

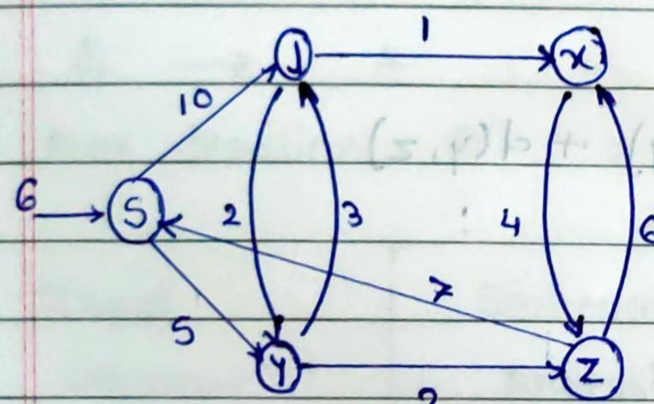
$= 3+4 = 7$

$d(A,C) = 3$

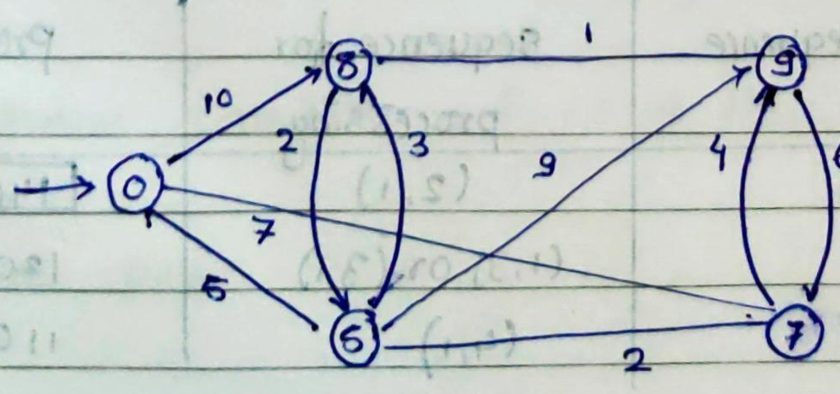
$d(A,D) = d(A,C) + d(C,B) + d(B,D)$
$= 3 + 4 + 2$
$= 9$

$d(A,E) = d(A,C) + d(C,E)$
$= 3 + 2 = 5$

2.



Q →	S	t	x	z	y
	0	∞	∞	∞	∞
	–	10	∞	∞	5
	–	8	14	7	–
	–	8	13	–	–
	–	–	9	–	–
	–	–	–	–	–



Shortest paths

$d(s, 1) = d(s, y) + d(y, 1)$
$= 5 + 3 = 8$

$d(s, y) = 5$
$d(s, x) = d(s, y) + d(y, 1) + d(1, x)$
$= 5 + 3 + 1$
$= 9$

$d(s, z) = d(s, y) + d(y, z)$
$= 5 + 2$
$= 7$

1/11/22 * **Job Scheduling with deadline**

1. Profit and deadline of jobs are given find optimal schedule with there maximum profit.

i	1	2	3	4
p_i ⇒	100	40	20	10
d_i ⇒	2	1	2	1

Max deadline

Input sequence	Sequence for processing	Profit
(1, 2)	(2, 1)	$\boxed{140}$
(1,3)	(1,3) or (3,1)	120
(1,4)	(4,1)	110

(2,3)	(2,3)	60
(2,4)	(2,4) / (4,2)	50
(3,4)	(4,3)	30

J_2	J_1
40	100

optimal schedule = (2,1) :

Max Profit = 140

2. i → 1 2 3 4

p_i → 15 10 15 13

d_i → 2 1 2 3

Max deadline = 3

Input Sequence	Sequence for processing	profit
(1,2,3)	(2,1,3) / (2,3,1)	75
(1,2,4)	(2,1,4)	38
(1,3,4)	(1,3,4) / (3,1,4)	$\boxed{78}$
(2,3,4)	(2,3,4)	73

optimal schedule = (1,3,4) / (3,1,4)

J_1	J_3	J_4
15	50	13

/

J_3	J_1	J_4
50	15	13

Max profit = 78

25
15
40
88

15
20
25
0
6

i →	1	2	3	4	5
P_i →	25	15	20	45	13
d_i →	3	1	1	2	3

Max deadline 3

Input sequence	Sequence for processing	profit
(1,2,3)	(2,3,1) / (3,2,1)	60
(1,2,4)	(2,4,1)	85
(1,2,5)	(1,5,2) / (5,1,2)	53
(1,3,5)	(3,1,5) / (3,5,1)	58
(1,4,5)	(4,1,5) / (4,5,1)	83

7/11/22

Dynamic Programming

1) Longest common Subsequence (LCS)
2) All paired shortest path - floyd Warshall Algorithm
3) Matrix chain multiplication (MCM)
4) Optimal Binary Search Tree (OBST)
5) 0/1 knapsack problem
6) Travelling Salesman Problem (TSP)
7) Multistage Graph
   1) forward approach
   2) Backword approach
8) Single Source shortest path - Bellman Ford algorithm

1. Longest Common Subsequence (LCS)

Q. Determine LCS of $x = (A, B, C, B, D, A, B)$ and $y = (B, D, C, ~~B~~, A, B, A)$

		x_i (0)	A (1)	B (2)	C (3)	B (4) ✓	D (5) ✓	A (6) ✓	B (7) ✓
0	y_j	0	0	0	0	0	0	0	0
✓ 1	B	0	↑ 0 (2)	↖ 1 (1)	← 1	↖ 1	← 1	← 1	↖ 1
✓ 2	D	0	↑ 0	↑ 0	↑ 1	↑ 1	↖ 2	← 2	← 2
3	C	0	↑ 0	↑ 1	↖ 2	← 2	↑ 2	↑ 2	↑ 2
✓ 4	A	0	↑ 1	↑ 1	↑ 2	↑ 2	↑ 2	↖ 3	← 3
✓ 5	B	0	↑ 1	↖ 2	↑ 2	↖ 3	← 3	↑ 3	↖ 4
6	A	0	↑	↑ 2	↑ 2	↑ 3	↑ 3	↖ 4	↑ 4

1) Rule

	x_i	A
y_j	0	0
A	0	↖

Adding one to the digonal point if element same

	x_i	A
y_i	0	0
A	0	↖1

↖

Rule 2 →

	x_i	B
y_i	0	0
A	0	↑ 0

↑ ∴ equal to less than

Rule 3 →

	x_j	B
y_i	0	0
A	1	←1

← element must be greater than

B < A ∴ 1 > 0

greater than

LCS (x, y) = (B, D, A, B)

Q Find LCS of X = Polynomial and y = exponential

y_j \ x_i	P	O	L	Y	N	O	M	I	A	L
E										
X										
P										
O										
N										
E										
N										

$x = (1, 0, 0, 1, 0, 1, 0, 1)$

$y = (0, 1, 0, 1, 1, 0, 1, 1, 0)$

	Xi	P	O	L	Y	N	O	M	I	A	L
yj	0	0	0	0	0	0	0	0	0	0	0
E	0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0
X	0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0	↑0
P	0	↖1	←1	←1	←1	←1	←1	←1	←1	←1	←1
O	0	↑1	↖2	←2	←2	←2	↖2	←2	←2	←2	←2
N	0	↑1	↑2	↑2	↑2	↖3	←3	←3	←3	←3	←3
E	0	↑1	↑2	↑2	↑2	↑3	↑3	↑	↑3	↑3	↑3
N	0	↑1	↑2	↑2	↑2	↑3	↑3	↑3	↑3	↑3	↑3
T	0	↑1	↑2	↑2	↑2	↑3	↑3	↑3	↑3	↑3	↑3
I	0	↑1	↑2	↑2	↑2	↑3	↑3	↑3	↖4	←4	←4
A	0	↑1	↑2	↑2	↑2	↑3	↑3	↑3	↑4	↖5	←5
L	0	↑1	↑2	↑2	↑2	↑3	↑3	↑3	↑4	↑5	↖6

P O N I A L

9/11/22

2] All pairs shortest path - Floyd warshall algo.

→ Floyd (W)

$D^{(0)} = W$

for k = 1 to n

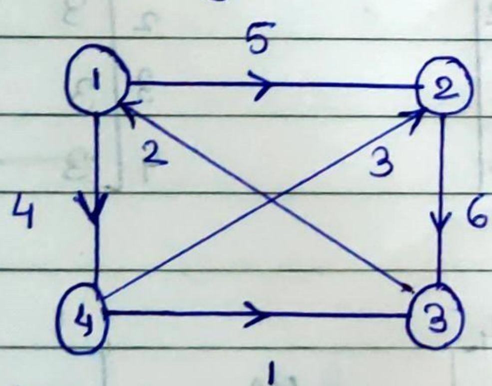
  for i = 1 to n

    for j = 1 to n

      $d_{ij}^{(k)} = \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$

return $D^{(n)}$

Qu. 1 find all pairs shortest path using floyd - warshall algorithm



$$D^{(0)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & \infty & 4 \\ 2 & \infty & 0 & 6 & \infty \\ 3 & 2 & \infty & 0 & \infty \\ 4 & \infty & 3 & 1 & 0 \end{array}$$

$$\pi^{(0)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & N & 1 & N & 1 \\ 2 & N & N & 2 & N \\ 3 & 3 & N & N & N \\ 4 & N & 4 & 4 & N \end{array}$$

$$\min(6, \infty+\infty) = (6, \infty) = 6$$

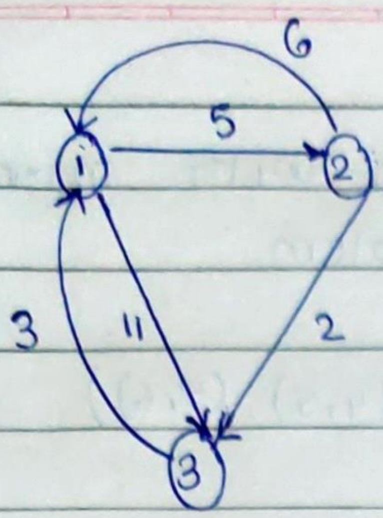
first row & first col^n as it is

$$D^{(1)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & \infty & 4 \\ 2 & \infty & 0 & 6 & \infty \\ 3 & 2 & 7 & 0 & 6 \\ 4 & \infty & 3 & 1 & 0 \end{array} \qquad \Pi^{(1)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & N & 1 & N & 1 \\ 2 & N & N & 2 & N \\ 3 & 3 & 1 & N & 1 \\ 4 & N & 4 & 4 & N \end{array}$$

$$D^{(2)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 11 & 4 \\ 2 & \infty & 0 & 6 & \infty \\ 3 & 2 & 7 & 0 & 6 \\ 4 & \infty & 3 & 1 & 0 \end{array} \qquad \Pi^{(2)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & N & 1 & 2 & 1 \\ 2 & N & N & 2 & N \\ 3 & 3 & 1 & N & 1 \\ 4 & N & 4 & 4 & N \end{array}$$

$$D^{(3)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 11 & 4 \\ 2 & 8 & 0 & 6 & 12 \\ 3 & 2 & 7 & 0 & 6 \\ 4 & 3 & 3 & 1 & 0 \end{array} \qquad \Pi^{(3)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & N & 1 & 2 & 1 \\ 2 & 3 & N & 2 & 1 \\ 3 & 3 & 1 & N & 1 \\ 4 & 3 & 4 & 4 & N \end{array}$$

$$D^{(4)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 5 & 5 & 4 \\ 2 & 8 & 0 & 6 & 12 \\ 3 & 2 & 7 & 0 & 6 \\ 4 & 3 & 3 & 1 & 0 \end{array} \qquad \Pi^{(4)} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & N & 1 & 4 & 1 \\ 2 & 3 & N & 2 & 1 \\ 3 & 3 & 1 & N & 1 \\ 4 & 3 & 4 & 4 & N \end{array}$$

② 

$$\Rightarrow D^{(0)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 5 & 11 \\ 2 & 6 & 0 & 2 \\ 3 & 3 & \infty & 0 \end{array} \qquad \Pi^{(0)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & N & 1 & 1 \\ 2 & 2 & N & 2 \\ 3 & 3 & N & N \end{array}$$

$$D^{(1)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 5 & 11 \\ 2 & 6 & 0 & 2 \\ 3 & 3 & 8 & 0 \end{array} \qquad \Pi^{(1)} = \begin{bmatrix} N & 1 & 1 \\ 2 & N & 2 \\ 3 & 1 & N \end{bmatrix}$$

$$D^{(2)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 5 & 7 \\ 2 & 6 & 0 & 2 \\ 3 & 3 & 8 & 0 \end{array} \qquad \Pi(2) = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & N & 1 & 2 \\ 2 & 2 & N & 2 \\ 3 & 3 & 1 & N \end{array}$$

$$D^{(3)} = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 5 & 7 \\ 2 & 5 & 0 & 2 \\ 3 & 3 & 8 & 0 \end{array} \qquad \Pi(3) = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & N & 1 & 2 \\ 2 & 3 & N & 2 \\ 3 & 3 & 1 & N \end{array}$$

1) Find optimal solution with maximum profit for 0/1 knapsack problem

$n = 4$, $M/W = 5$

$(w_i, p_i) = ((2,3), (3,4), (4,5), (5,6))$

p_i	w_i	$i \downarrow$ / $w_i \rightarrow$	0	1	2	3	4	(w_i, p_i)
		0	0	0	0	0	0	
3	2	1	0	0	3	3	3	(2,3)
4	3	2	0	0	3	4	7	(3,4)
5	4	3	0	0	3	5	7	(4,5)
6	5	4	0	0	3	5	7	(5,6)

4+0, 3   5+0, 4   8+   4+0, 3   max = 3+

Max profit $= B(n, w)$

$= B(4, 5)$

$= 7$

Optimal Solution = (1, 1, 0, 0)

2) $n = 3$, $M = 6$

$w_i \rightarrow 2, 3, 3$

$p_i \rightarrow 1, 2, 4$

GOODLUCK Page No.
Date

profit

Matrix chain multiplication

$A_1 \; A_2$
$\downarrow \quad \downarrow$
$10 \times 15 \quad 15 \times 20$

1) $(A_1 A_2) A_3$

2) $A(A_2, A_3)$

1) $d = (10, 100, 5, 50)$

$A_1 = 10 \times 100$

$A_2 = 100 \times 5$

$A_3 = 5 \times 50$

① $(A_1, A_2) A_3$

$(A_1, A_2) = p \times q \times r = 10 \times 100 \times 5$
$= 5000$

$\downarrow$
10×5

$(A_1 A_2) A_3 = 10 \times 5 \times 50$
$= 2500$

$10 \times 5 \quad 5 \times 50$

$= 5000 + 2500$

$= 7500$

② $A_1(A_2 \times A_3)$

$A_2 \times A_3 = 100 \times 5 \times 50$
$= 2500$

$A_1 (A_2 \times A_3) = 10 \times 100 \times 50$

$= 50{,}000$

$\underset{10 \times 100}{A_1} \quad \underset{100 \times 50}{(A_2 \times A_3)}$

$= 25000 + 50{,}000$

$= 75{,}000$

1. Find optimal parenthisisation for the matrices whose dimension is $d = (10, 100, 5, 50)$
$\qquad d_0 \; d_1 \; d_2 \; d_3$

=)

$j = [\, m[i,j] \,]$

	1	2	3	
	0	5000	7500	1 ← S = 2
		0	25000	2 ← S = 1
			0	3 ← S = 0

$A_1 = 10 \times 100$

$A_2 = 100 \times 5$

$A_3 = 5 \times 50$

for $S = 0$

for $i = 1$ to $n - S$ $\quad (\because n - S = 3 - 0 = 3)$

$\quad i = 1$ to 3

$m(i,j) = 0$

For $S = 1$

for $i = 1$ to 2 $\qquad (n - S = 3 - 1 = 2$

$m(i, i+1) \qquad = d_{i-1} d_i d_{i+1}$

$i = 1$

$$m(1,2) = d_0 d_1 d_2 = 10 \times 100 \times 5 = 5000$$

$i = 2$

$$m(2,3) = d_1 d_2 d_3 = 100 \times 5 \times 50 = 25{,}000$$

For $s = 2$

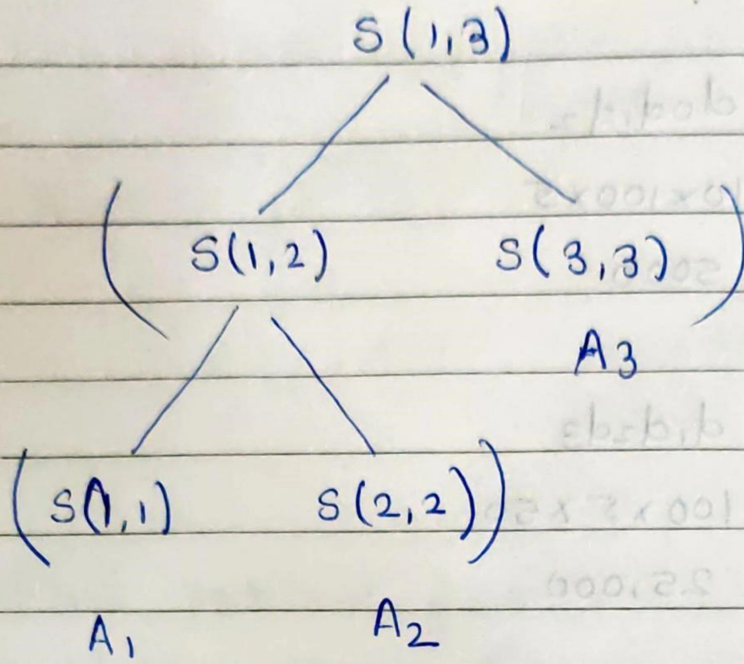
for $i = 1$ to 1

$$m(i, i+s) = \min_{i \leq k < i+s} \left[m(i,k) + m(k+1, i+s) + d_{i-1} d_k d_{i+s} \right]$$

$i = 1$

$$m(1,3) = \min_{1 \leq k < 3} \left[\begin{array}{l} m(1,1) + m(2,3) + d_0 \times d_1 \times d_3 \\ \quad = 0 + 25{,}000 + (10 \times 100 \times 50) \\ \quad = 75000 \\ m(1,2) + m(3,3) + d_0 \times d_2 \times d_3 \\ \quad = 5000 + 0 + (10 \times 5 \times 50) \\ \quad = 7500 \end{array} \right]$$

Smallest $\leftarrow$



$\rightarrow ((A_1 . A_2) A_3)$

GOODLUCK Page No.
Date

Unit-4.

Backtracking strategy

1) n-Queens problem
2) sum of Subsets
3) Graph coloring
4) Hamiltonian cycle

UNit 5

NP-hard and NP-complete Problem

- Non-deterministic algo.
- NP hard & NP-complete
- decision and optimization problem
- clique
- Polynomial Reduction
- Cook's Theorem
- Graph based problem on NP-principle