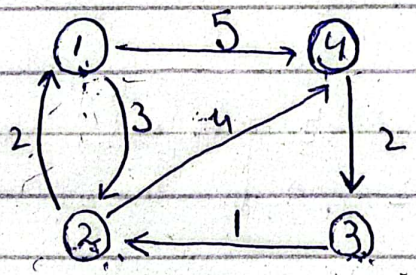
Analysis of Algorithm.

Dynamic programming.

→ Divide & conquer vs Dynamic programming.
→ Optimize solution.
→ Dynamic programming Algorithms.
  → Floyd Warshall Algorithm.
  → Febonocce series.
  → Binomial coefficient.
  → String Method.
  → Matrix chain Multiplicating

* Floyd Warshall Algorithm.



A_0

	1	2	3	4
1	0	3	∞	5
2	2	0	∞	4
3	∞	1	0	∞
4	∞	∞	2	0

In matric 2 Row 1 of A_0 & column 1 will be same

A_1

	1	2	3	4
1	0	3	∞	5
2	2	0	9	4
3	∞	1	0	∞
4	∞	∞	2	0

We choose the path passing through 1 or direct & choose minimum.

A_2

	1	2	3	4
1	0	3	9	5
2	2	0	9	4
3	3	1	0	5
4	∞	∞	2	0

A_3

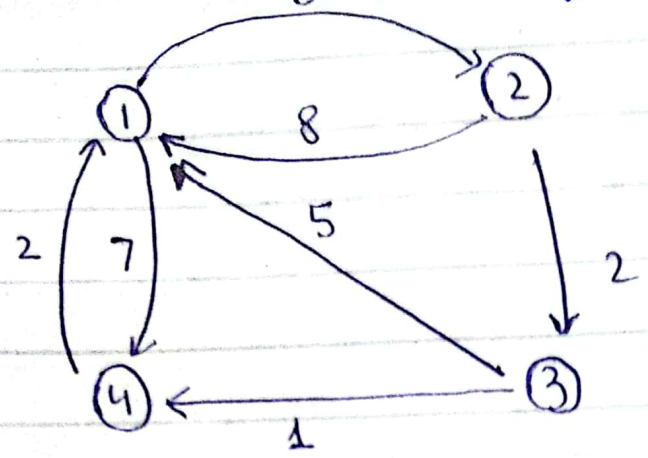
	1	2	3	4
1	0	3	9	5
2	2	0	9	4
3	3	1	0	5
4	5	3	2	0

A_4

	1	2	3	4
1	0	3	7	5
2	2	0	6	4
3	3	1	0	5
4	5	3	2	0

(using previous matrix in last matrix if there is no direct or indirect path)

Task:



$$A_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & \infty & 7 \\ 8 & 0 & 2 & \infty \\ 5 & \infty & 0 & 1 \\ 2 & \infty & \infty & 0 \end{bmatrix} \end{array}$$

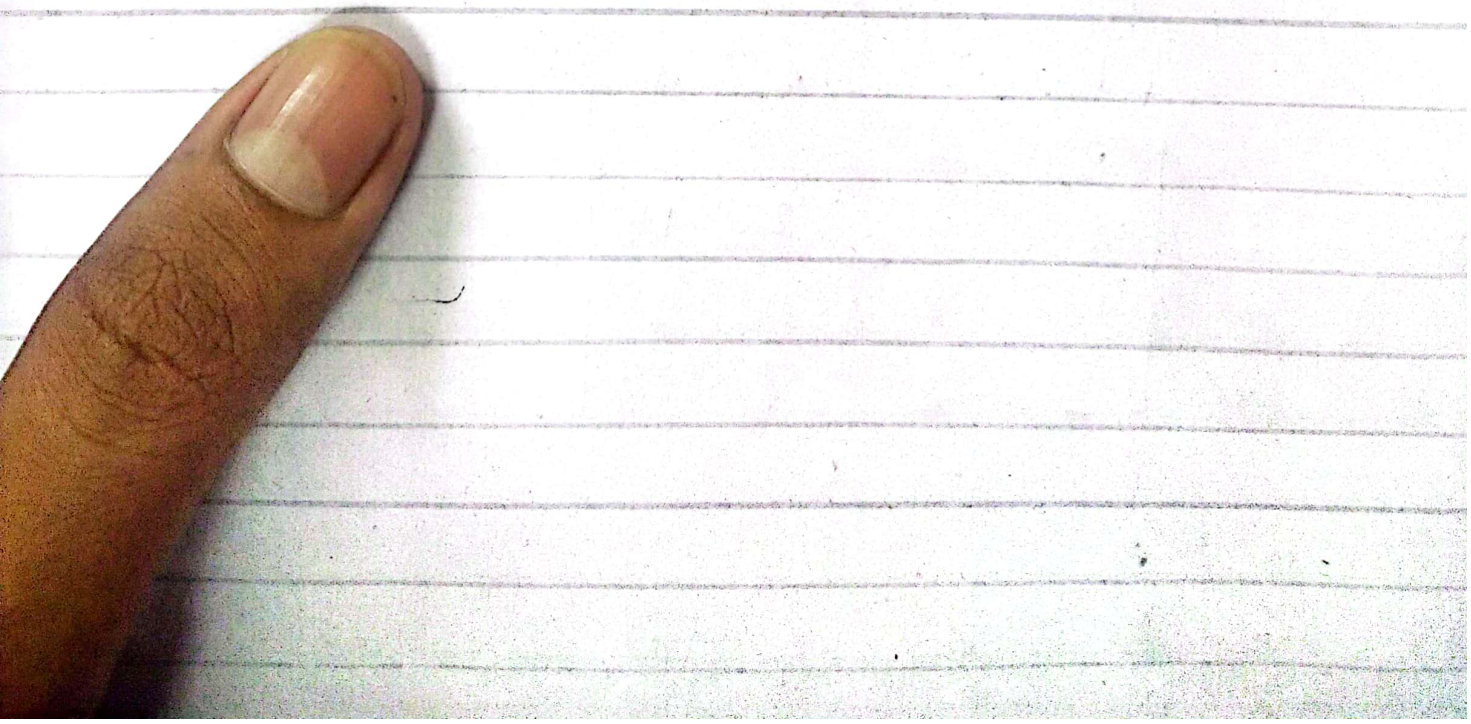
$$A_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & \infty & 7 \\ 8 & 0 & 2 & 15 \\ 5 & 8 & 0 & 1 \\ 2 & 5 & 7 & 0 \end{bmatrix} \end{array}$$

$$A_2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & 5 & 7 \\ 8 & 0 & 2 & 15 \\ 5 & 8 & 0 & 1 \\ 2 & 5 & \infty & 0 \end{bmatrix} \end{array}$$

$$A_3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & 5 & 7 \\ 7 & 0 & 2 & 3 \\ 5 & 8 & 0 & 1 \\ 2 & \infty & \infty & 6 \end{bmatrix} \end{array}$$

$A_4 =$

	1	2	3	4
1	0	3	∞	7
2	8	0	2	3
3	3	7	0	1
4	2	∞	∞	0



Dynamic programming.

→ Floyd Warshall algorithm.
→ Fabonacchi series.
→ Binomial coefficient.
  → Concept.
  → Mathmatical procedure.
  → Binomial coefficient and Dynamic programming
  → Example & exercise.

Binomial coefficient.

A binomial coefficient is an algorithm expression of the form a+b that we multiply as.

$(a+b)^2 : a^2+2ab+b^2$

$(a+b)^3 : a^3+3a^2b+3ab^2+b^3$

The Mathmatical formula for Binomial coefficient is.

$$c[n,k] = {}^nC_r = \frac{n!}{r!(n-r)!}$$

${}^5C_3$ using Dynamic programming.

n r→	0	1	2	3
0	1	X		
1	1	1	X	
2	1	2	1	X
3	1	3	3	1
4	1	4	6	4
5	1	5	10	10

$c[i,j] = c[i-1,j] + c[i-1,j-1]$

# i=0 to n
  j=0 to min (i,r)

∴ whenever r > n place X there.
∴ whenever r = n blank the place.

Dynamic programming:
→ Matrix chain Multiplication.

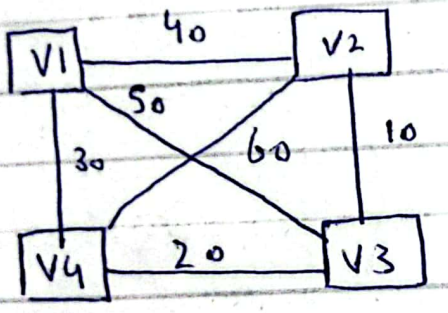
Greed algorithm:
→ concept
→ Optimized solution
→ Application.

Syed ubaid

→ Greedy algorithm → Kruskal's algorithm.
→ Prims Algorithm
→ Dijekstra Algorithm.

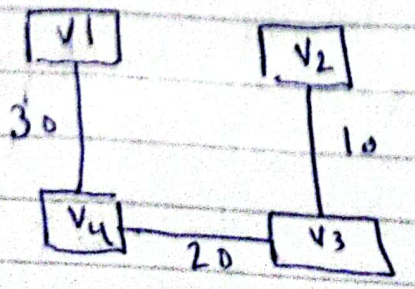
13-5-23

→ Minimum spanning Tree [MST]
→ Finding MST using Kruskal Algorithm.
→ Example & exercise



Edges	weight.
(v_2, v_3)	10
(v_3, v_4)	20
(v_1, v_4)	30
(v_1, v_2)	40
(v_1, v_3)	50
(v_2, v_4)	60

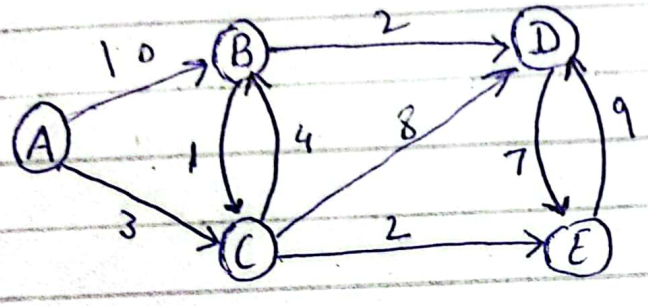
Action	UV = $\{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}\}$
Added	$\{\{v_1\}, \{v_2, v_3\}, \{v_4\}\}$
Added	$\{\{v_1\}, \{v_2, v_3, v_4\}\}$
Added	$\{\{v_1, v_2, v_3, v_4\}\}$
Reject	Already added
Reject	Already added
Reject	Already added



10+30+20=60

Dijekstra Algorithm:-

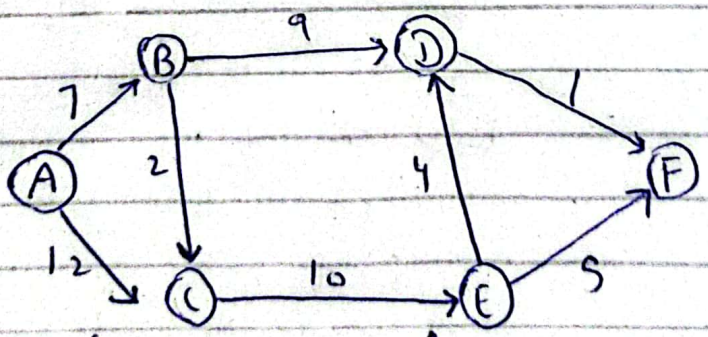
→ Concept.
→ Shortest Path finder
→ Example & exercise.



$S = \{A, C, E, B, D\}$

Q	A	B	C	D	E
A	0	∞	∞	∞	∞
C		10	3	∞	∞
E		7		11	5
B				11	
D					

Example.

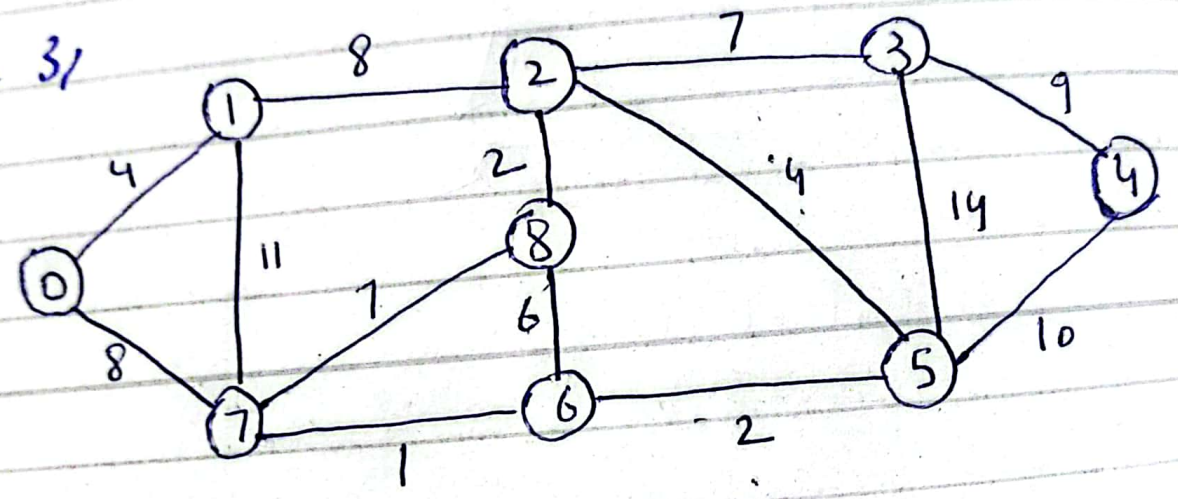


S = {A, B, C

Q	A	B	C	D	E	F
A	0	∞	∞	∞	∞	∞
B		7	12	16	∞	∞
D			9	16	∞	∞

A	B	C	D	E	F
			16	19	∞
				19	17
				19	

Example 3/



S = {0, 1, 7, 6, 5, 2, 8, 3, 4}

Q	0	1	2	3	4	5	6	7	8
	0	∞	∞	∞	∞	∞	∞	∞	∞
		4	12	∞	∞	∞	∞	8	∞
			12	∞	∞	∞	∞	8	∞
				12	∞	∞	9	∞	15
				12	∞	∞	11		15
				12	∞	∞			15
				12	25	21			19
					19	21			
					19	21			
						21			