

Q1 Adjacency Matrix

	A	B	C	D	E	F	G	H	I
A	0	1	1	1	0	0	0	0	0
B	1	0	1	0	0	1	0	1	0
C	1	1	0	1	1	1	0	0	0
D	1	0	1	0	1	0	0	0	1
E	0	0	1	1	0	1	1	0	0
F	0	1	1	0	1	0	1	1	0
G	0	0	0	0	1	1	0	1	1
H	0	1	0	0	0	1	1	0	1
I	0	0	0	1	0	0	1	1	0

Q2 Find the shortest path from A to all other vertices using Dijkstra's Algorithm

$$1. A \begin{cases} A-B: 22 \leftarrow \\ A-C: 3 \leftarrow \\ A-D: 12 \leftarrow \end{cases}$$

$$4. B \begin{cases} B-C: 57X \\ B-F: 58X \\ B-H: 56 \leftarrow \end{cases}$$

$$2. C \begin{cases} C-D: 13X \\ C-E: 74X \\ C-F: 51 \leftarrow \end{cases}$$

$$3. D \begin{cases} D-E: 45 \leftarrow \\ D-I: 42 \leftarrow \end{cases}$$

$$5. E \begin{cases} E-F: 63X \\ E-G: 68 \leftarrow \end{cases}$$

$$6. F \begin{cases} F-G: 90X \\ F-H: 75X \end{cases}$$

$$8. G \begin{cases} G-H: 93X \\ G-I: 89X \end{cases}$$

$$7. H \begin{cases} H-I: 75X \end{cases}$$

(4). I

dis:

	A	B	C	D	E	F	G	H	I
0	22	3	12	45	51	68	56	42	42

path:	A	B	C	D	E	F	G	H	I
	{ }	{A-B}	{A-C}	{A-D}	{A-D-E}	{A-C-F}	{A-D-E-G}	{A-B-H}	{A-D-I}

Q3 What is the time complexity?

Q4 Find a minimum spanning tree using Kruskal's Algorithm

$$L = \left[\begin{array}{l} (C, D)^4, (A, C)^3, (E, F)^{18}, (H, I)^{19}, \text{~~(D, E)^{20}}~~, (G, I)^{21}, (A, B)^{22}, (E, G)^{23}, (F, H)^{24}, (G, H)^{25}, \\ (D, I)^{30}, (D, E)^{31}, (B, H)^{34}, (B, C)^{35}, (B, F)^{36}, (F, G)^{37}, (C, F)^{42}, (C, E)^{45} \end{array} \right]$$

$$\text{Union-Find: } \{A\}, \{B\}, \{C\}, \{D\}, \{E\}, \{F\}, \{G\}, \{H\}, \{I\} \Rightarrow n = 9$$

$$\text{MST edges} = 9 - 1 = 8$$

$$\text{pick}(C, D) \Rightarrow \text{Find}(C) \neq \text{Find}(D) \cdot \text{Union}(C, D)$$

$$\{A\}, \{B\}, \{C, D\}, \{E\}, \{F\}, \{G\}, \{H\}, \{I\}$$

$$\text{pick}(A, C) \Rightarrow \text{Find}(A) \neq \text{Find}(C) \cdot \text{Union}(A, C)$$

$$\{A, C, D\}, \{B\}, \{E\}, \{F\}, \{G\}, \{H\}, \{I\}$$

$$\text{pick}(E, F) \Rightarrow \text{Find}(E) \neq \text{Find}(F) \cdot \text{Union}(E, F)$$

$$\{A, C, D\}, \{B\}, \{E, F\}, \{G\}, \{H\}, \{I\}$$

$$\text{pick}(H, I) \Rightarrow \text{Find}(H) \neq \text{Find}(I) \cdot \text{Union}(H, I)$$

$$\{A, C, D\}, \{B\}, \{E, F\}, \{G\}, \{H, I\}$$

$$\text{pick}(G, I) \Rightarrow \text{Find}(G) \neq \text{Find}(I) \cdot \text{Union}(G, I)$$

$$\{A, C, D\}, \{B\}, \{E, F\}, \{G, H, I\}$$

$$\text{pick}(A, B) \Rightarrow \text{Find}(A) \neq \text{Find}(B) \cdot \text{Union}(A, B)$$

$$\{A, B, C, D\}, \{E, F\}, \{G, H, I\}$$

$$\text{pick}(E, G) \Rightarrow \text{Find}(E) \neq \text{Find}(G) \cdot \text{Union}(E, G)$$

$$\{A, B, C, D\}, \{E, F, G, H, I\}$$

$$\text{pick}(D, I) \Rightarrow \text{Find}(D) \neq \text{Find}(I) \cdot \text{Union}(D, I)$$

$$\{A, B, C, D, E, F, G, H, I\} \quad // \text{ we can stop here since we have } n-1 \text{ edges}$$

Q5. What is the time complexity?

Q6. DAG adjacency Matrix

	P	Q	R	S	T	U
P	0	1	0	6	7	6
Q	0	0	1	4	0	0
R	0	0	0	2	0	1
S	0	0	0	0	3	2
T	0	0	0	0	0	2
U	0	0	0	0	0	0

rows(i)
columns(j)

Wt of $i \rightarrow j$ is 0 if there is no edges

Q7. Shortest Path From P to U using DAG algorithm

dist:

P	Q	R	S	T	U
0	1	2	3	4	7

Path:

P	Q	R	U	S	T
{}	P-Q	P-Q-R	P-Q-R-U	P-Q-R-S	P-T

Final Result:

• $dist[U] = 3$

• Shortest path: P-Q-R-U

Q8. DAG's algorithm Time complexity

• $O(V+E)$ or $O(n+m)$

Q9. Can you use Dijkstra's algorithm to find the shortest path from P to U?

• Yes, because all vertices have positive weights and the graph is a directed graph with non-negative weights

Q10. Shortest Path using Dijkstra's algorithm

• P-Q-R-U

Time complexity: $O(V^2)$