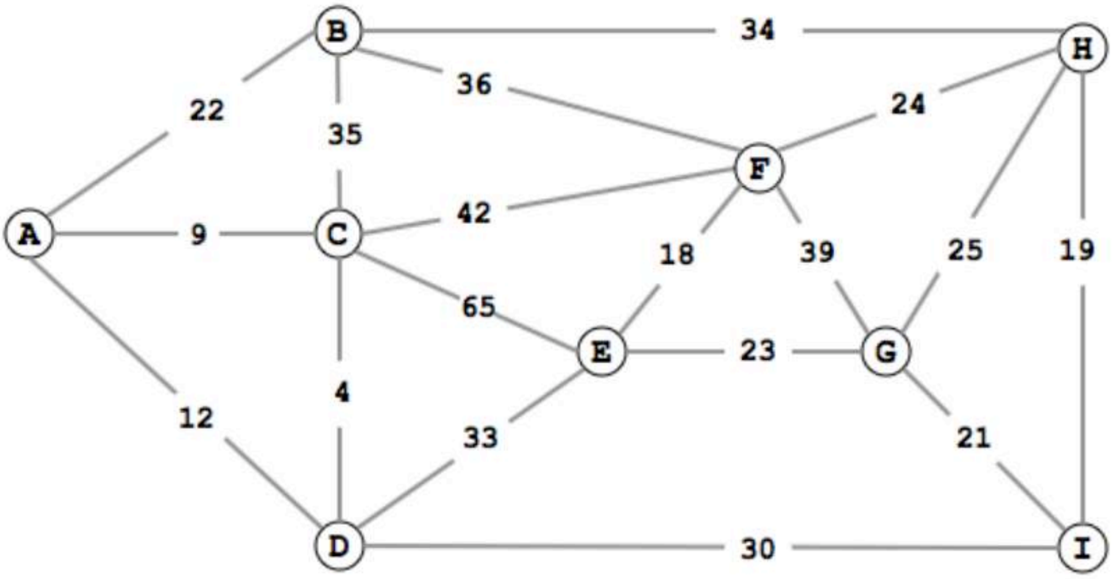
Course: Algorithm
Prof. Prem Nair
Student: Binh Van Tran
ID: 986648
Homework: Lab 14

1. **Question 1 –** What is the adjacency matrix of the weighted graph G = (V, E) displayed below



	A	B	C	D	E	F	G	H	I
A	0	22	9	12	0	0	0	0	0
B	22	0	35	0	0	36	0	34	0
C	9	35	0	4	65	42	0	0	0
D	12	0	4	0	33	0	0	0	30
E	0	0	65	33	0	18	23	0	0
F	0	36	42	0	18	0	39	24	0
G	0	0	0	0	23	39	0	25	21
H	0	34	0	0	0	24	25	0	19
I	0	0	0	30	0	0	21	19	0

2. **Question 2 –** Find the shortest path from A to all other vertices using Dijkstra's algorithm

- $A[A] = 0$; $X = \{A\}$

$wt(A,B) = 0 + 22 = 22$

$wt(A,C) = 0 + 9 = 9$ ← min weight

$wt(A,D) = 0 + 12 = 12$

- $A[C] = 9$; $X = \{\underline{A}, \underline{C}\}$

$wt(A,B) = 22$

$wt(A,D) = 12$ ← min

$A[C] + wt(C,B) = 9 + 35 = 44$

$A[C] + wt(C,D) = 9 + 4 = 13$

$A[C] + wt(C,F) = 9 + 42 = 51$

$A[C] + wt(C,E) = 9 + 65 = 74$

- $A[D] = 12$; $X = \{A, C, D\}$

$wt(A,B) = 22$ ← min

$A[C] + wt(C,B) = 44$

$A[C] + wt(C,F) = 51$

$A[C] + wt(C,E) = 74$

$A[D] + wt(D,E) = 12 + 33 = 45$

$A[D] + wt(D,I) = 12 + 30 = 42$

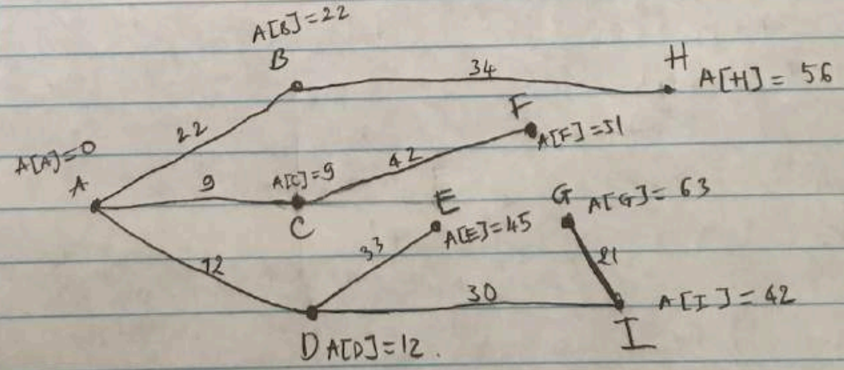
- $A[B] = 22$; $X = \{A, C, D, B\}$

$A[B] + wt(B,H) = 22 + 34 = 56$

$A[B] + wt(B,F) = 22 + 36 = 58$

A[C] + wt(C,F) = 51
A[C] + wt(C,E) = 74
A[D] + wt(D,E) = 45
A[D] + wt(D,I) = 42 ← min
• A[I] = 42 ; X = {A, C, D, B, I}
A[B] + wt(B,H) = 56
A[B] + wt(B,F) = 58
A[C] + wt(C, F) = 51
A[C] + wt(C,E) = 74
A[D] + wt(D,E) = 45 ← min
A[I] + wt(I, G) = 42 + 21 = 63
A[I] + wt(I,H) = 42+19 = 61

• A[E] = ~~51~~ 45 ; X = {A, C, D, B, I, E}
A[B] + wt(B, H) = 56
A[B] + wt(B, F) = 58
A[C] + wt(C, F) = 51 ← min
A[I] + wt(I, G) = 63
A[I] + wt(I, H) = 61
A[E] + wt(E, F) = 45 + 18 = 63
A[E] + wt(E, G) = 45 + 23 = 68
• A[F] = 51 ; X = {A, C, D, B, I, E, F}
A[B] + wt(B, H) = 56 ← min
A[I] + wt(I, G) = 63
A[I] + wt(I, H) = 61
A[E] + wt(E, G) = 68
A[F] + wt(F, G) = 51 + 39 = 90
A[F] + wt(F, H) = 51 + 24 = 75
• A[H] = 56 ; X = {A, C, D, B, I, E, F, H}
A[I] + wt(I, G) = 63 ← min
A[E] + wt(E, G) = 68
A[F] + wt(F, G) = 90
A[H] + wt(H, G) = 56 + 25 = 81
• A[G] = 63 ; $X = \{A, \underbrace{C, D, B, I, E, F, H, G}_{\text{all vertices ; stop}}\}$



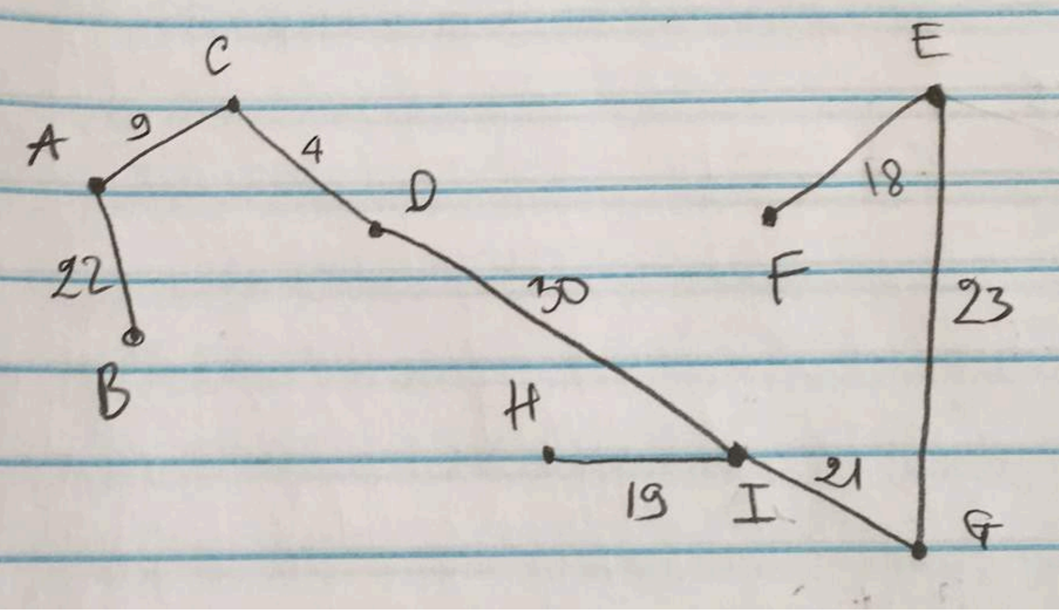
3. **Question 3 –** What is the time complexity using Dijkstra

   We are using adjacency matrix to represent a graph.
   Let say, v is the no of vertices. Due to the exhaustive search of edges is made in each iteration, the running time should be in *$O(v^2)$*
   In this specific problem, v = 9 so we can easily calculate the running time in the worst case is $9^2 = 81$

4. **Question 4** – Find a minimum spanning tree using Kruskal's Algorithm

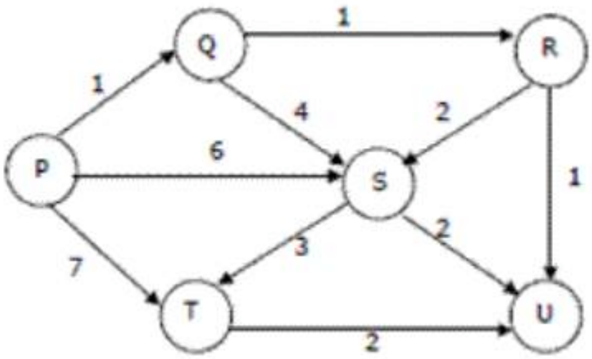
Finding Minimum Spanning tree

(C, D) = 4 ✓
(A, C) = 9 ✓
(A, D) = 12 ✓
(E, F) = 18 ✓
(H, I) = 19 ✓
(G, I) = 21 ✓
(A, B) = 22 ✓
(E, G) = 23 ✓
(F, H) = 24 ✓
(G,H) = 25 ✓
(D, I) = 30 ✓
(D, E) = 33
(B, H) = 34
(B, C) = 35
(B, F) = 36
(F, G) = 39
(C, F) = 42
(C,E) = 65



5. **Question 5** – What is time complexity using Kruskal
   m is number of edges, n is number of vertices
   Time to sort the edges: *O(mlogm)*
   Loop from 0 to n – 1 edges to build the clusters: *O(mlogn)*.
   n > m + 1
   m <= n^2 ➔ *logm* < $\log n^2$ ➔ log*m* < *2logn*
   So, time complexity is *O(mlogn)*

6. **Question 6** – What is the adjacency matrix of the weighted directed acyclic graph G = (V, E) below



Topological order: P→Q→R→S→T→U

	P	Q	R	S	T	U
P	0	1	0	6	7	0
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

7. **Question 7** – Find the shortest path from P to U using Dijkstra's (dynamic programming)

D[P] = 0
D[Q] = 1
D[R] = 2
D[S] = min $(\underbrace{D[Q] + wt(Q,S)}_{5},\ \underbrace{D[P] + wt(P,S)}_{6},\ \underbrace{D[R] + wt(Q,R)}_{4})$
= min(5 , 6 ,)
= 4
D[T] = min $(\underbrace{D[P] + wt(P,T)}_{0+7},\ D[S] + wt(S,T))$
= min(0 + 7 , 4 + 3)
= 7
D[U] = min (D[T] + wt(T,U), D[S] + wt(S,U), D[R] + wt(R,U))
= min(9 , 6 , 3)
= 3
Shortest path from P → U : minimal cost is 3.

$P \xrightarrow{1} Q \xrightarrow{1} R \xrightarrow{1} U$

8. **Question 8** – What is time complexity

   Time complexity is *O(n + m),* n is number of vertices, m is number of edges

9. **Question 9** – Can you use Dijkstra's algorithm (Slide 12) to find the shortest path from P to U?

   We cannot use because Dijkstra's algorithm (no dynamic programming) is only for finding shortest path between two vertices in undirected graph.
   But in this problem, we can use because this graph has no negative weight.

10. **Question 10** – Find the shortest path form P to U using Dijkstra (no dynamic programming)

- A[P] = 0 ; X = {P}
  - wt(P, Q) = 1 ← min
  - wt(P, S) = 6
  - wt(P, T) = 7
- A[Q] = 1 ; X = {P, Q}
  - A[P] + wt(P, S) = 6
  - A[P] + wt(P, T) = 7
  - A[Q] + wt(Q, S) = 1 + 4 = 5
  - A[Q] + wt(Q, R) = 1+1 ← min
- A[R] = 2 ; X = {P, Q, R}
  - A[P] + wt(P, S) = 6
  - A[P] + wt(P, T) = 7
  - A[Q] + wt(Q, S) = 5
  - A[R] + wt(R, S) = 4
  - A[R] + wt(R, U) = 3 ← min
- A[U] = 3 ; X = {P, Q, R, U}

⇒ Shortest path P → U ; cost = 3

$P \xrightarrow{1} Q \xrightarrow{1} R \xrightarrow{1} U$