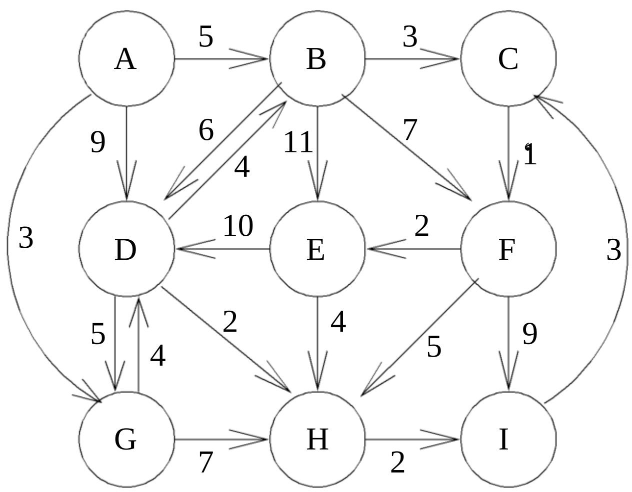
Problem Set 6

Name: Subhankari Mishra
Assigned: July 1
Due: July 8

Problem 1

Trace the execution of Dijkstra's algorithm on the following graph, taking A as the starting vertex. Show the successive states of the array D[i] and of the set of vertices whose distance has been determined.



Ans: S = {A}
D = {0, ∞, ∞, ∞, ∞, ∞, ∞, ∞, ∞}
A, B, C, D, E, F, G, H, I

W <- A
D[A] + C[A -> B] = 5
D[A] + C[A -> D] = 9
D[A] + C[A -> G] = 3

P =

A	B	C	D	E	F	G	H	I
-	A		A			A		

D = {0, 5, ∞, 9, ∞, ∞, 3, ∞, ∞}
A, B, C, D, E, F, G, H, I
W <- G
S = {A, G}
D[G] + C[G -> D] = 7
D[G] + C[G -> H] = 10
D = {0, 5, ∞, 7, ∞, ∞, 3, 10, ∞}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A		G			A	G	

W <- B
S = {A, G, B}
D[B] + C[B -> C] = 8
D[B] + C[B -> D] = 11
D[B] + C[B -> E] = 16

D[B] + C[B -> F] = 12
D = {0, 5, 8, 7, 16, 12, 3, 10, ∞}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	B	A	G	

W <- D
S = {A, G, B, D}
D[D] + C[D -> B] = 11
D[D] + C[D -> H] = 9
D[D] + C[D -> G] = 12
D = {0, 5, 8, 7, 16, 12, 3, 9, ∞}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	B	A	D	

W <- C
S = {A, G, B, D, C}
D[C] + C[C -> F] = 9
D = {0, 5, 8, 7, 16, 9, 3, 9, ∞}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	C	A	D	

W <- F
S = {A, G, B, D, C, F}
D[F] + C[F -> E] = 11
D[F] + C[F -> H] = 14
D[F] + C[F -> I] = 18

D = {0, 5, 8, 7, 16, 9, 3, 9, 18}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	C	A	D	F

W <- H
S = {A, G, B, D, C, F,H}
D[H] + C[H -> I] = 11

D = {0, 5, 8, 7, 16, 9, 3, 9, 11}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	C	A	D	H

W <- I
S = {A, G, B, D, C, F, H, I}
D [I] + C [I -> C] = 14

D = {0, 5, 8, 7, 16, 9, 3, 9, 11}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	C	A	D	H

W <- E
S = {A, G, B, D, C, F, H, I, E}
D [E] + C [E -> D] = 26
D [E] + C [E -> H] = 20

D = {0, 5, 8, 7, 16, 9, 3, 9, 12}
A, B, C, D, E, F, G, H, I

P =

A	B	C	D	E	F	G	H	I
-	A	B	G	B	C	A	D	H

Problem 2 (Siegel Ex. 8.2)

Write an enhancement to the Floyd-Warshall algorithm that saves, in Intermediate[i,j], the last vertex before j on the shortest path from i to j. Notice that proper initialization makes the algorithm easier.

```
Ans:Procedure F-W(n, EdgeCost[1..n,1..n];;PathCost[1..n,1..n],kIntermediate[1..n,1..n]);
        PathCost[*,*] <= EdgeCost[*,*];
        Forall i <- 1 to n do
                Forall j <- 1 to n do
                        If there is an edge from i to j then kIntermediate[i,j] <- i
                        Else kIntermediate[i,j] <- nil endif.
                Endfor;
        Endfor;
        For k <- 1 to n do
                Forall i <- 1 to n do
                        Forall j <- 1 to n do
                                If PathCost[i,j] > PathCost[i,k] + PathCost[k,j] then
                                        PathCost[i,j] <- PathCost[i,k] + PathCost[k,j];
                                        kIntermediate[i,j] <- kIntermediate[k,j];
                                endif;
                        endfor;
                endfor;
        endfor
end_F-W;
```

Problem 3

Consider the following problem. You are given a directed graph G with two disjoint subsets P and Q. A path is considered invalid if it goes first through a vertex in P and then through a vertex in Q. For example P and Q may be points in enemy countries, and Q may prohibit travelers whose passport has a visa to P. Or in an epidemic of a communicable disease, one may want to block people who have been through Q from entering B.

For example, in the graph in problem 1, if U = {B, C} and V = {D, E} then the path A→B→F →E→H →I is invalid, because it first goes through U at vertex B and then later goes through V, at vertex E. The path A→D→B→F →I is valid, because D in V comes before B in U.

Describe how the method of cloning can be used to find the optimal valid path.

Ans: Create a super graph H consisting of the graph G' and G'' where G' has all the vertices of graph G and the edges U_1->V_1 for all U_1 and V_1 that belong to P with the same cost as in G.
Similarly, G'' has all the vertices of graph G and the edges U_2->V_2 for all U_2 and V_2 that belong to Q with the same cost as in G.
Draw the edges from U_1->V_2 for the original edges U -> V for all U, V where $U \in P$ and $V \in Q$ and assign cost as ∞.
Now draw the edges from U_2->V_1 for the original edges U->V for all U,V where $U \in Q$ and $V \in P$ and assign the original cost of U->V in graph G.
Run the Floyd-Warshall algorithm for the super graph H with 2N vertices to get the optimal valid path.