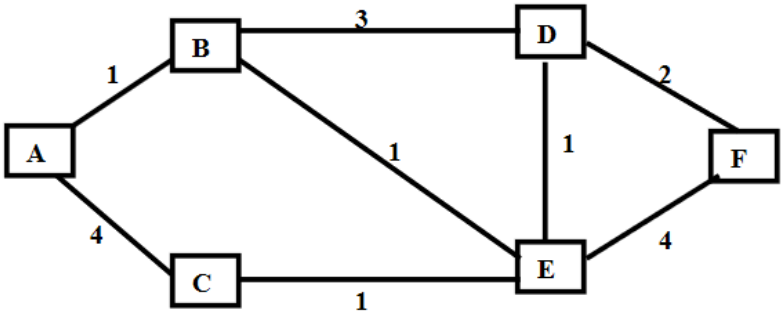
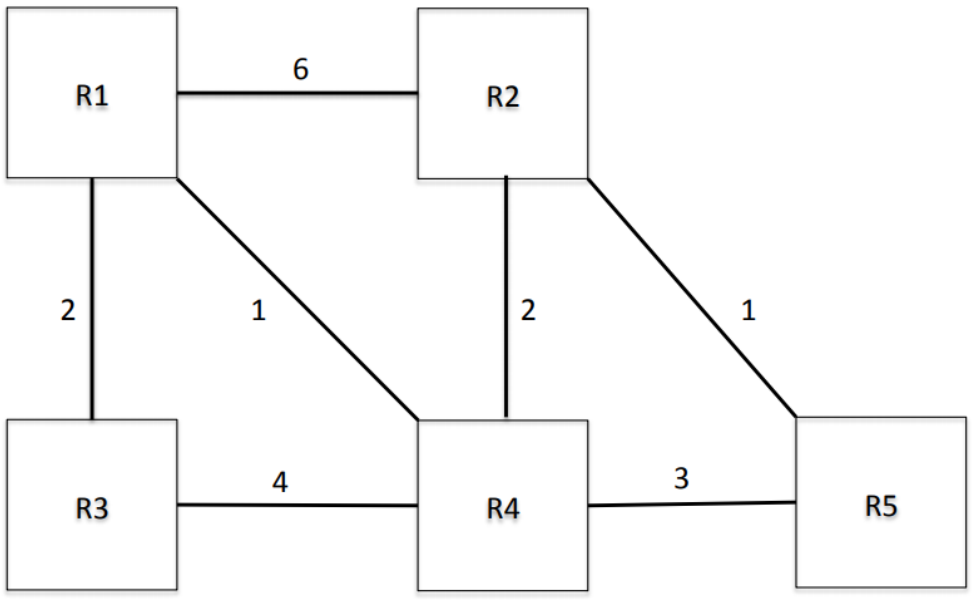
Sample Problems on Routing Algorithms
(No Solutions will be posted)

1. Consider the following computer network where each node represents a router and the edge label is the corresponding link cost. All links are bi-directional. Use Dijkstra algorithm to find the shortest path from router "A" to every other router in the network.



2. Consider the following network:

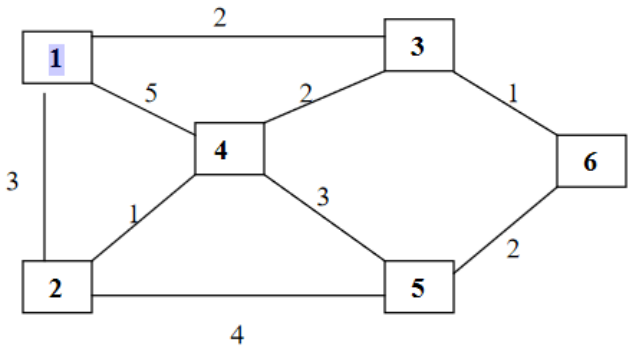
Use the Dijkstra Algorithm (The shortest Spanning Tree) and the Bellman-Ford (Distance Vector) algorithms to find the shortest distance from R3 to all other routers.



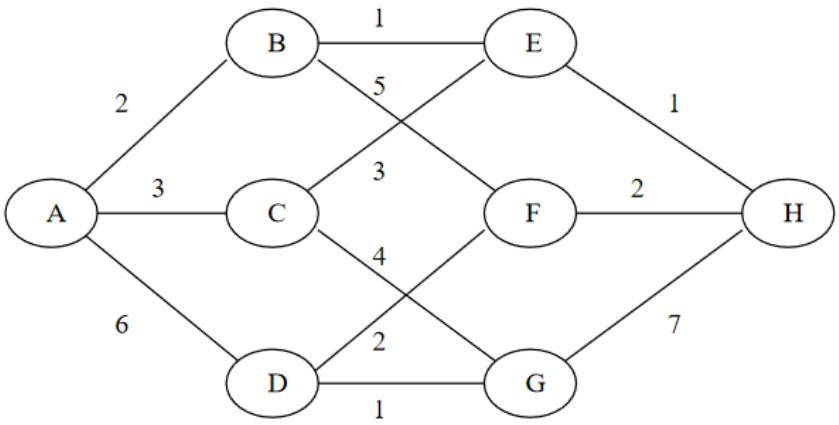
3. For the network shown below, assume that the Bellman-Ford routing algorithm is used. Assume that initially each router knows the distance to its neighbors. Illustrate the following:

- Initial routing table at node "1"
- Final routing table at node"1"
- Show clearly, how the algorithm would react if there is a failure in the link connecting nodes "3" and node "6"

Note: the routing table at R_1 should have three columns, namely: Destination, Cost and Next Hop.



4. Consider the following computer network where each node represents a router and the edge label is the corresponding link cost. Use Dijkstra algorithm to find the shortest path from router B to every other router in the network. Show your work step-by-step. Now suppose the link between router D and router F is down. Describe how the link-state protocol handles it. What would be the shortest path between router B and router G?



5. We have five routers A through E. Their forwarding tables after the RIP stabilizes are shown below. Assume all links are of cost 1.

a. A packet originated at A and destined to E. What path will it take?

b. A packet originated at C and destined to D. What path will it take?

c. Can you sketch a possible network consistent with the tables?

Forwarding Table for A

Destination	Cost	Next Hop
A	0	-
B	1	B
C	2	B
D	1	D
E	2	D

Forwarding Table for B

Destination	Cost	Next Hop
A	1	A
B	0	-
C	1	C
D	1	D
E	1	E

Forwarding Table for C

Destination	Cost	Next Hop
A	2	B
B	1	B
C	0	-
D	2	E
E	1	E

Forwarding Table for D

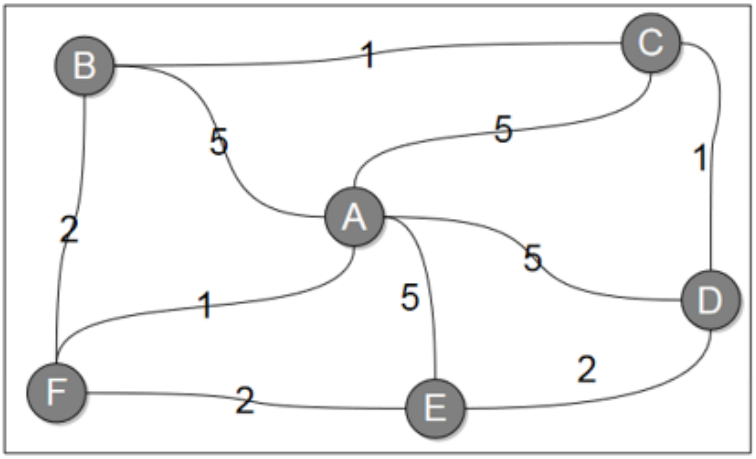
Destination	Cost	Next Hop
A	1	A
B	1	B
C	2	B
D	0	-
E	1	E

Forwarding Table for E

Destination	Cost	Next Hop
A	2	D
B	1	B
C	1	C
D	1	D
E	0	-

6.

Complete the following table, using Dijkstra's algorithm.
Compute the shortest path from node A to all network nodes.
Note: Possible ties **must be** broken in favor of the leftmost column.



Step	*N'*	*D(B),p(B)*	*D(C),p(C)*	*D(D),p(D)*	*D(E),p(E)*	*D(F),p(F)*
0	A					
1						
2						
3						
4						

7. We use CIDR. Without using longest prefix matching, a forwarding table looks like this.

If we use longest prefix matching, we can combine a few entries together. What is a table with a minimum number of entries that still be able to forward packet correctly?

Prefix	Outgoing Interface
128.0.0.0/11	eth1
128.16.0.0/12	eth1
128.24.0.0/12	eth2
128.32.0.0/12	eth2
128.40.0.0/12	eth1
128.48.0.0/11	eth1
128.64.0.0/9	eth0
128.128.0.0/10	eth0
128.160.0.0/11	eth1
128.176.0.0/11	eth0
128.192.0.0/9	eth0
default	eth3

$\Rightarrow$

Prefix	Outgoing Interface
default	eth3