King Fahd University of Petroleum and Minerals
College of Computer Sciences and Engineering
Department of Computer Engineering

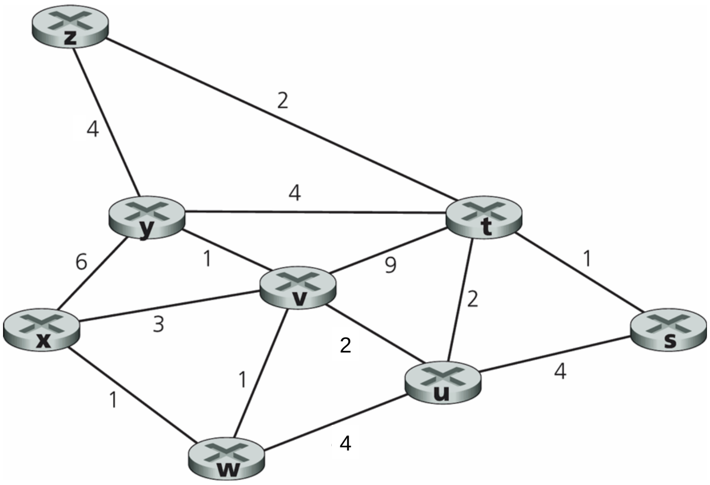
COE 344 – Computer Networks (T152)

Homework # 04 (*due date & time: Tuesday 12/04/2016 during class period*)

Late homework submission will NOT be accepted

***** Show all your work. No credit will be given if work is not shown! *****

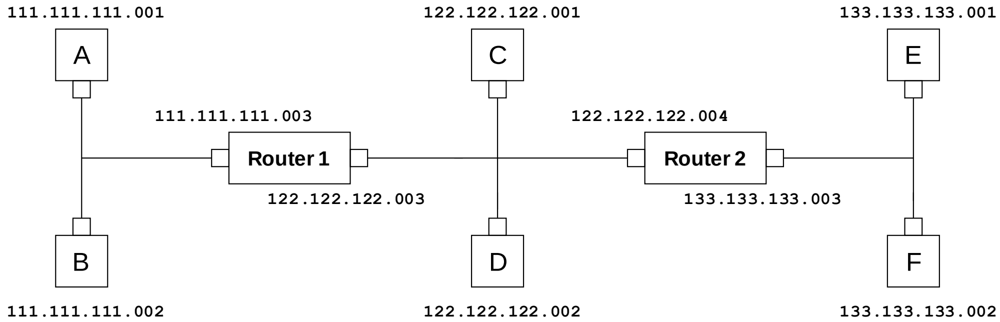
Problem # 1 (24 points):
Consider the following network.



With the indicated link costs, use *Dijkstra*'s shortest-path algorithm, as discussed in class, to compute the shortest path from ***y*** to all network nodes using the table given below.

N'	*D(s),p(s)*	*D(t),p(t)*	*D(u),p(u)*	*D(v),p(v)*	*D(w),p(w)*	*D(x),p(x)*	*D(z),p(z)*

<u>Problem # 2 (24 points):</u> Consider the following IP-based network with the assigned IP addresses as shown. For each of the following cases, complete the table regarding the datagram as it is forwarded from the source to the destination.



1. Assume that host ***F*** sends an IP datagram to host ***D***.

Source IP address	Destination IP address	Receiving interface IP address that was passed down to the Data Link layer to be used for forwarding

2. Assume that host ***F*** sends an IP datagram to host ***E***.

Source IP address	Destination IP address	Receiving interface IP address that was passed down to the Data Link layer to be used for forwarding

3. Assume that host ***F*** sends an IP datagram to host ***A***.

Source IP address	Destination IP address	Receiving interface IP address that was passed down to the Data Link layer to be used for forwarding

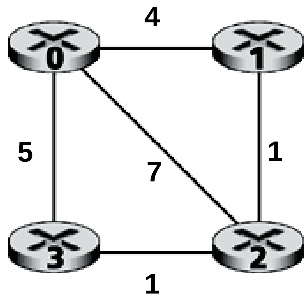
Problem # 3 (10 points): Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 224.1.18/24. Also suppose that Subnet 1 is required to support at least 60 interfaces, Subnet 2 is to support at least 11 interfaces, and Subnet 3 is to support at least 91 interfaces. Provide three network addresses (of the form *a.b.c.d/x*) that satisfy these constraints.

Problem # 4 (18 points): Consider a datagram network using 32-bit host addresses. Suppose a router has five links, numbered 0 through 4, and datagrams are to be forwarded to the link interfaces as follows:

Destination Address Range	**Link Interface**
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 00001111 11111111	1
11100000 01000000 00010000 00000000 through 11100000 01000000 11111111 11111111	2
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	3
otherwise	4

a. Provide an equivalent forwarding table that translates each of the destination address ranges into a single prefix of the form *a.b.c.d/x* along with the associated link interface. The forwarding table should have six entries (4 for the four given ranges, and 2 for the "otherwise" range).
b. Determine the appropriate link interface for the following datagrams with destination addresses:
   i. 11100000 01010000 11000011 00111100
   ii. 11100001 10000000 00010001 01110111
   iii. 11100000 01000000 00001000 11111111
   iv. 11001000 10010001 01010001 01010101

Problem # 5 (24 points): Consider the following network.



Suppose that the link cost *c*(0,1) has changed from 4 to 1, re-compute the distance tables for nodes 0, 1, 2, and 3 after each iteration of a synchronous version of the distance vector algorithm using as many of the following tables as needed. Note that the current tables' values *prior* to the link cost change are as shown in the leftmost column of the tables.

cost to
from

D^0	0	1	2	3
0	0	4	5	5
1	4	0	1	2
2	5	1	0	1
3	5	2	1	0

cost to
from

D^0	0	1	2	3
0				
1				
2				
3				

cost to
from

D^0	0	1	2	3
0				
1				
2				
3				

cost to
from

D^0	0	1	2	3
0				
1				
2				
3				

cost to
from

D^0	0	1	2	3
0				
1				
2				
3				

cost to
from

D^1	0	1	2	3
0	0	4	5	5
1	4	0	1	2
2	5	1	0	1
3	∞	∞	∞	∞

cost to
from

D^1	0	1	2	3
0				
1				
2				
3				

cost to
from

D^1	0	1	2	3
0				
1				
2				
3				

cost to
from

D^1	0	1	2	3
0				
1				
2				
3				

cost to
from

D^1	0	1	2	3
0				
1				
2				
3				

cost to
from

D^2	0	1	2	3
0	0	4	5	5
1	4	0	1	2
2	5	1	0	1
3	5	2	1	0

cost to
from

D^2	0	1	2	3
0				
1				
2				
3				

cost to
from

D^2	0	1	2	3
0				
1				
2				
3				

cost to
from

D^2	0	1	2	3
0				
1				
2				
3				

cost to
from

D^2	0	1	2	3
0				
1				
2				
3				

cost to
from

D^3	0	1	2	3
0	0	4	5	5
1	∞	∞	∞	∞
2	5	1	0	1
3	5	2	1	0

cost to
from

D^3	0	1	2	3
0				
1				
2				
3				

cost to
from

D^3	0	1	2	3
0				
1				
2				
3				

cost to
from

D^3	0	1	2	3
0				
1				
2				
3				

cost to
from

D^3	0	1	2	3
0				
1				
2				
3				