ECE 665, Computer Algorithms, Spring 2005
Homework 4
Due: Thursday, April 28

1. Chapter 7 Exercises, **C-7.7** (telephone network)

2. Chapter 7 Exercises, **C-7.8** (max bandwidth computation)

3. Develop a **Minimum Spanning Tree** algorithm using Kruskal's method for an arbitrary graph (not necessarily connected). Illustrate it with a simple example.

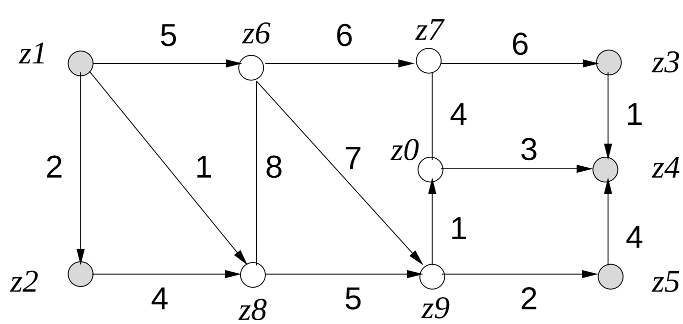
4. **Assignment Problem**
The following matrix M specifies the possible assignment (matching) of workers to jobs, where $M(i,j) = 1$ indicates that worker i can perform job j.

	1	2	3	4	5	6
1				1		
2		1	1			1
3				1		
4			1		1	
5	1					1
6						1

Find the solution that maximizes the matching of workers with jobs. Each worker can perform at most one job and each job can be done by at most one worker. The cost of an assignment of a worker to a job is 1.

Formulate the problem as a network flow problem. Which algorithm will you use to solve it? Show all the steps of your algorithm and prove that your solution is optimum.

5. **Multi-commodity Flow Problem**.
Nodes z_1 and z_2 represent factories which are capable of producing certain material at rates of 7 and 5 units, respectively. Nodes z_3, z_4 and z_5 represent depots which are capable of receiving the units at respective rates of 6, 4 and 3.



At what rate should z_1 and z_2 produce and over what routes should the material be shipped to *maximize* the product flow? Each route has a maximum capacity for transporting the

material, shown in the graph.
Assume that the capacity of the undirected edges, (z_6, z_8) and (z_7, z_0)), specifies the total amount of commodity that can flow through that edge in *both* directions. Make sure to model this constraint on the graph.

Describe the model you use to solve this problem and show all your work.

6. **Programming Assignment**

   Implement a computer program to solve the **Flight Scheduling** problem described in the textbook (problem C-7.9). Explain how you model the problem as graph algorithm.

   Consider the following airline schedule (which is one of the inputs to your program). Assume that the connection times are equal to 1 hour for each airport.

(SFO, LAX) - (2:00, 3:00)	(SFO, DFW) - (1:30, 4:30)
(SFO, ORD) - (0.30, 4:30)	(LAX, DFW) - (4:00, 6:00)
(DFW, MIA) - (8:00, 10:00)	(DFW, ORD) - (6:15, 9:15)
(ORD, BOS) - (10:30, 13:30)	(ORD, JFK) - (10:15, 13:15)
(MIA, JFK) - (12:00, 15:00)	(JFK, BOS) - (14:50, 15:50)
(MIA, BOS) - (11:30, 14:30)	(SFO, BOS) - (0:45, 6:45)
(LAX, MIA) - (5:00, 9:00)	(LAX, SFO) - (17:00, 18:00)
(BOS, MIA) - (10:00, 13:00)	(BOS, JFK) - (11:00, 12:00)
(BOS, ORD) - (11:00, 14:00)	(BOS, SFO) - (7:30, 13:30)
(JFK, MIA) - (13:15, 16:15)	(JFK, ORD) - (8:00, 11:00)
(ORD, SFO) - (12:00, 16:00)	(ORD, DFW) - (16:15, 19:15)
(MIA, DFW) - (14:00, 16:00)	(MIA, LAX) - (11:00, 15:00)
(DFW, LAX) - (21:00, 23:00)	(DFW, SFO) - (18:00, 21:00).

   Find schedule(s) with the *earliest arrival time* for the following user data. If more than one solution exists, your program should select the one with minimum number of connections.

   - From SFO to JFK, starting at 00:30.
   - From BOS to LAX, starting at 09:30.
   - From MIA to SFO, starting at 12:00

   The input data with the airline schedule is a list of connections in the following format:

   airport_start departure_time airport_terminal arrival_time

   with time given in 24-hour system. For example:

   SFO 1400 LAX 1500
   SFO 1330 DFW 1630
   etc.

   The exact input format for this airline schedule is available on the homework web page.

   Show the solution (give a printout of your running program). Send the TA a working link to an executable, clearly indicating the operating system. Give clear instructions how to run it.