COMP 360 - Winter 2016 - Midterm

Name:

1. 169 students are in need of emergency treatment. Each of the 169 students requires a transfusion of one unit of whole blood. The clinic has supplies of 170 units of whole blood. The number of units of blood available in each of the four major blood groups and the distribution of patients among the groups is summarized below.

Blood type	A	B	O	AB
Supply	46	34	45	45
Demand	39	38	42	50

Type A patients can only receive type A or O; type B patients can receive only type B or O; type O patients can receive only type O; and type AB patients can receive any of the four types.

Solution: Put two vertices for every blood type, one corresponding to supply and one to demand. Add an edge of infinite capacity from a supply blood type vetex to a demand blood type vertex if the corresponding patients can receive that supply type. For example an inifinite capacity edge goes from O^{supply} to A^{demand}. Connect the source to supply vertices with capacities equal to the number of available supplies (e.g. sA^{supply} with capacity 46, and an edge from the demand to sink with the number of demands (e.g. $A^{demand}t$ has capacity 39.)

   (a) (15 points) Give a max flow formulation that determines a distribution that satisfies the demands of a maximum number of patients (Explain why the solution to your formulation corresponds to the problem). Draw the directed graph and put the edge capacity above each edge. Your network should have 10 vertices: a source (named s), a supply node for each of the four blood types (named a, b, o, ab), a demand node for each blood type (named a', b', o', ab'), and a sink (named t).
   Solution: One student dies.

   (b) (10 points) Solve the maximum flow problem using the Ford-Fulkerson algorithm. Do the first augmentation on the path $s - b - ab' - t$. List each of the augmenting paths below, and state the amount by which the path increases the current value of the flow. Also, write the final flow values on each edge in the network above.
   Solution: straightforward.

   (c) (10 points) Find a minimum cut in the network above, i.e. list the vertices on the source side of the cut.
   Solution: straightforward.

2. (20 points) Prove that for every flow network, there exists an execution of the Ford-Fulkerson algorithm that finds the maximum flow using at most m augmentations, where m is the number of the edges of the network.

Solution: To prove this, we apply the following algorithm to decompose a maximum flow f into a set of flow-paths:

- Repeat the following two steps until there is no such s, t-path:
- Find an s, t-path P with positive flow, i.e., $f(e) > 0$ for all $e \in P$.
- Let Δ be the minimum value of the flow f on edges of P. Decrease the flow f on each edge $e \in P$ by Δ.

In every iteration the value of f on at least one edge becomes 0. Hence the number of iterations is at most m. Using these paths as augmenting paths (with augmenting value Δ) leads to the desired result.

3. (15 points) A factory produces two products (A, B) using two machines (M, N). Product A requires processes on both machines, but Product B can be produced on either machine M or N. The following table shows the processing times (in minutes) on each machine

Product	Machine N	Machine M
A	15	18
B	20	25

Each machine works for 8 hours every day. Due to marketing limitations, the number of Product A sold must be at least the number of Product B sold. When sold, each unit of Product A and B contributes to profit \$16 and \$20, respectively. Write a linear program to maximize the daily profit. What is the optimal solution to this linear program?

Solution:

$$\begin{array}{ll} \max & 16x_1 + 20x_{2,N} + 20x_{2,M} \\ \text{s.t.} & \\ & 15x_1 + 20x_{2,N} \leq 8 \times 60 \\ & 18x_1 + 25x_{2,M} \leq 8 \times 60 \\ & x_1 - x_{2,M} - x_{2,N} \geq 0 \\ & x_1, x_{2,M}, x_{2,N} \geq 0 \end{array}$$

4. (15 Points) Give an example of a linear program that is infeasible and its dual is also infeasible.

$$\begin{array}{ll} \max & x_1 + 2x_2 - x_3 \\ \text{s.t.} & x_1 \leq 1 \\ & -x_1 \leq -2 \\ & x_2 - x_3 \leq 0 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$

Dual:

$$\begin{array}{ll} \max & y_1 - 2y_2 \\ \text{s.t.} & y_1 - y_2 \geq 1 \\ & y_3 \geq 2 \\ & -y_3 \geq -1 \\ & y_1, y_2, y_3 \geq 0 \end{array}$$