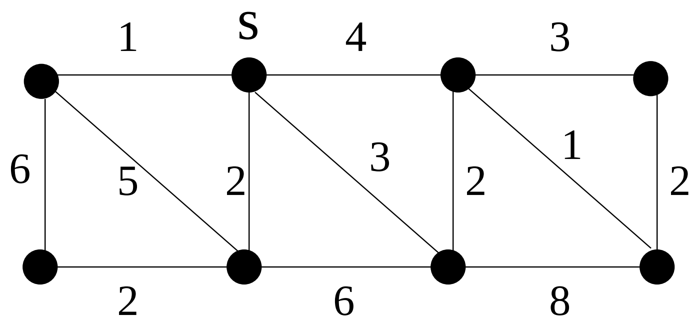
CS140: Exam 2 Name ______________________________

11/15/00

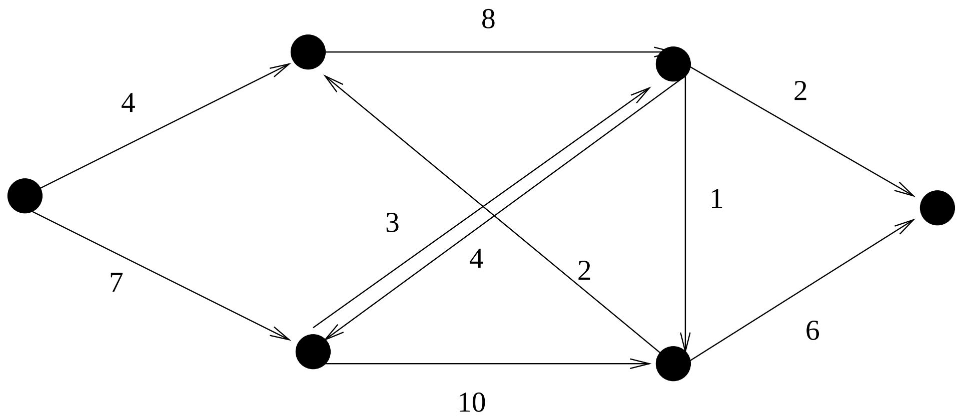
This exam consists of 4 problems and 5 pages. Make sure your exam is complete. Write your name above.

Your answers will be judged for clarity, conciseness, and correctness. The space provided should be adequate for your solutions. Bounds and algorithms should be, respectively, as tight and as fast as possible. You may use a one page cheat sheet but no text or other notes during this exam.

1. (15 points) Find a smallest path tree for the vertex s in the following graph.



2. (20 points) Find the maximum flow in the following network. Clearly label each edge of the network with the flow on that edge. Give a minimum cut of the network



3. (25 points) M is an $n \times m$ matrix of distinct integers in which the rows are sorted in ascending order (smallest at left) as well as the columns (smallest at top). The numbering convention we'll follow is that $M(1,1)$ is the upper, left element of the matrix and $M(n,m)$ is the lower right element.

   (a) (1 points) Construct a sample 3×4 matrix that is sorted as described.

   (b) (2 points) Give the indices of the elements of M that are guaranteed to be strictly smaller than the element $M(k,\ell)$.

   (c) (2 points) Give the indices of the elements of M that are guaranteed to be strictly larger than the element $M(k,\ell)$.

(d) (20 points) Design an algorithm to search for an integer x in M. The algorithm can access M through queries of the form "How does $M(k, \ell)$ compare to x?" which is answered (a) "$M(k, \ell) = x$," (b) "$M(k, \ell) > x$," or (c) "$M(k, \ell) < x$." Your algorithm should return the location of x in M (the indices) or report that x is not in M.

To receive full credit your algorithm should make at most $m + n - 1$ queries in worst case. To receive half-credit give an $O(n + m)$ algorithm.

Hint: Find a first query such that a (b) or (c) response allows you to eliminate one row or one column from consideration.

4. (40 points) Given a parenthesis-free expression over the integers and $\{+, -, *\}$ we want to parenthesize the expression so as to maximize its value. For example, the expression $-2 \; + \; 5 \; * \; 3 \; - \; 10$ can be parenthesized as $-2 + ((5 * 3) - \; 10)$, which evaluates to 3. It can also be parenthesized as $((-2 + 5) * 3) - 10$, which evaluates to -1.

   (a) (5 points) Parenthesize the following expressions so as to maximize their values.

      i. $10 \; * \; -3 \; + 5$

      ii. $3 \; - \; 4 \; - \; 2 \; * \; 5 \; + \; -1$

   (b) (35 points) Design an algorithm that takes as input an expression $v_1 o_1 v_2 o_2 \ldots o_{n-1} v_n$, where each v_i is an integer and each o_i is an operator from $\{+, -, *\}$, and parenthesizes it so as to maximize its values. Hint 1: Strengthen the inductive hypothesis. Hint 2: Dynamic programming.