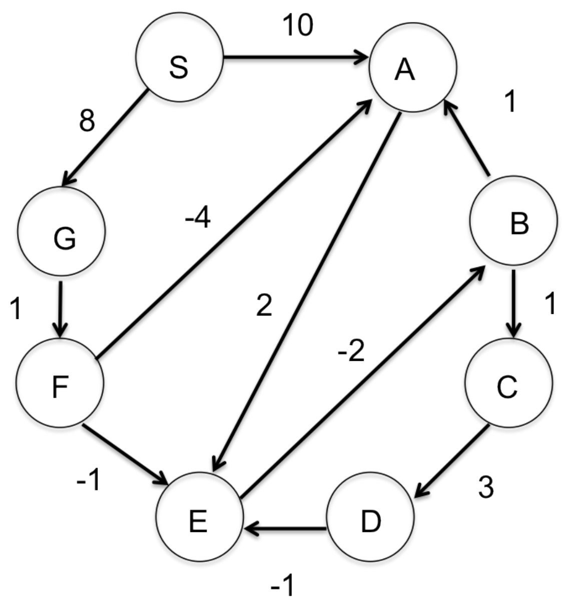
CS483 - Final Exam Pratice
Prof. Jana Košecká

1. (10) Find the shortest distances from the vertex S to all the other vertices. Show the intermediate distance values of all the nodes at each iteration of the algorithm.



2. (10) Give a liner time algorithm that takes as input directed acyclic graph $G = (V, E)$ and two vertices s and t and returns the number of simple paths from s to t. Your algorithm should just count the paths not list them. Simple path is a path with no repeated vertices or edges. The algorithm should be linear in $O(V + E)$ and exploit the fact that the graph is acyclic.

3. (10) How to cut steel rod into pieces to maximize the revenue you can get ? Each cut is free and rods are always integral number of inches.

   Input: A length n and table of prices p_i for $i = 1 \ldots n$.
   Output: The maximum revenue obtainable for rods whose lengths sum to n, computed as sum of the prices of individual rods.

   For example given following prices and lengths:

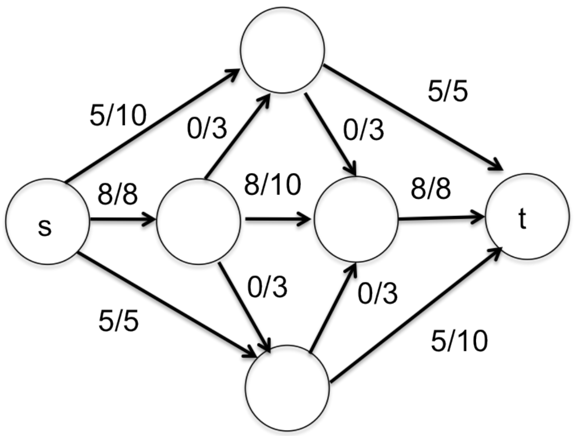
length i	1	2	3	4	5	6	7	8
price p_i	1	5	8	9	10	17	17	20

   To cut a rod of length 4, the best way to cut it is in to 2-ich pieces, getting revenue $p_2 + p_2 = 5 + 5 = 10$.

   Design an algorithm to solve the above problem. Give the running time complexity of your algorithm.

4. (10) Given a sequence o n real numbers A(1), ..., A(n) determine a contiguous subsequence A(i) ... A(j) for which the sum of elements in the subsequence is maximized. (Note that the array can have negative numbers). The algorithm should run in $O(n)$ time.

5. (10) The figure below shows a network on which flow has been computed. Each edge is labelled by flow/capacity on that edge.
(a) what is the value of the flow ?
(b) Is this a maximum (s-t) flow ? Argue why yes or no.
(c) Find a minimum (s-t) cut in the flow network and and say what's its capacity.



6. (15) Are the following statements TRUE or FALSE. Justify in few sentences.

(3) A greedy algorithm for a problem can never give an optimal solution on all inputs.

(3) Every NP-complete problem is in NP.

(3) If a problem A is polynomially reducible to problem B and problem A is NP-complete then problem B is also NP-complete.

(3) The approximate solutions to NP-complete problems can never give an optimal answer.

(3) Suppose you have a graph G and its Minimum Weight Spanning Tree T. Now replace each edge cost with c_e with c_e^2. T must still be a minimum spanning tree for this new instance. If it is true give a short explanation, if it is false give a counter example.

7. (10) Let 2-CNF-SAT be a set of satisfiable boolean formulas in CNF with exactly 2 literals per clause. Show that 2-CNF-SAT $\in$ P. Suggest an efficient algorithm. (Hint: Observe that $x \vee y$ is equivalent to $\neg x \rightarrow y$). Reduce 2-CNF-SAT to a problem in a directed graph that is efficiently solvable.