CS570

Analysis of Algorithms
Spring 2015
Exam II

Name: ________________________

Student ID: ___________________

Email Address:_________________

_____Check if DEN Student

	Maximum	Received
Problem 1	20	
Problem 2	20	
Problem 3	20	
Problem 4	20	
Problem 5	20	
Total	100	

Instructions:
1. This is a 2-hr exam. Closed book and notes
2. If a description to an algorithm or a proof is required please limit your description or proof to within 150 words, preferably not exceeding the space allotted for that question.
3. No space other than the pages in the exam booklet will be scanned for grading.
4. If you require an additional page for a question, you can use the extra page provided within this booklet. However please indicate clearly that you are continuing the solution on the additional page.

1) 20 pts

Mark the following statements as **TRUE** or **FALSE**. No need to provide any justification.

[TRUE/FALSE]

A flow network with unique edge capacities has a unique min cut.

[TRUE/FALSE]

If a problem can be solved by dynamic programming, then it can always be solved by exhaustive search (Brute Force).

[TRUE/FALSE]

A divide and conquer algorithm acting on an input size of n can have a lower bound less than Θ(n log n).

[TRUE/FALSE]

If a flow in a network has a cycle, this flow is not a valid flow.

[TRUE/FALSE]

In the divide and conquer algorithm to compute the closest pair among a given set of points on the plane, if the sorted order of the points on both X and Y axis are given as an added input, then the running time of the algorithm improves to O(n).

[TRUE/FALSE]

In a flow network, an edge that goes straight from *s* to *t* is always saturated when maximum *s* - *t* flow is reached.

[TRUE/FALSE]

The Bellman-Ford algorithm always fails to find the shortest path between two nodes in a graph if there is a negative cycle present in the graph.

[TRUE/FALSE]

If f is a max s-t flow of a flow network G with source s and sink t, then the capacity of the min s-t cut in the residual graph G_f is 0.

[TRUE/FALSE]

In a dynamic programming solution, the space requirement is always at least as big as the number of unique sub problems.

[TRUE/FALSE]

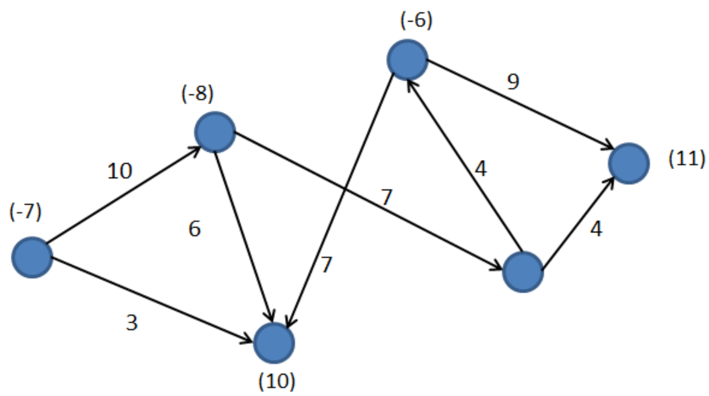
Decreasing the capacity of an edge that belongs to a min cut in a flow network may not result in decreasing the maximum flow.

2) 20 pts

A city is located at one node x in an undirected network $G = (V, E)$ of channels. There is a big river beside the network. In the rainy season, the flood from the river flows into the network through a set of nodes Y. Assume that the flood can only flow along the edges of the network. Let c_{uv} (integer value) represent the minimum effort (counted in certain effort unit) of building a dam to stop the flood flowing through edge (u, v). The goal is to determine the minimum total effort of building dams to prevent the flood from reaching the city. Give a pseudo-polynomial time algorithm to solve this problem. Justify your algorithm.

3) 20 pts

The following graph G is an instance of a circulation problem with demands. The edge weights represent capacities and the node weights (in parantheses) represent demands. A negative demand implies source.



(i) Transform this graph into an instance of max-flow problem.

(ii) Now, assume that each edge of G has a constraint of lower bound of 1 unit, i.e., one unit must flow along all edges. Find the new instance of max-flow problem that includes the lower bound constarint.

4) 20 pts

There is a series of activities lined up one after the other, $J_1, J_2, \dots, J_n$. The i^{th} activity takes T_i units of time, and you are given M_i amount of money for it. Also for the i^{th} activity, you are given N_i, which is the number of immediately following activities that you cannot take if you perform that i^{th} activity. Give a dynamic programming solution to maximize the amount of money one can make in T units of time. Note that an activity has to be completed in order to make any money on it. State the runtime of your algorithm.

5) 20 pts

A polygon is called convex if all of its internal angles are less than 180°and none of the edges cross each other. We represent a convex polygon as an array V with n elements, where each element represents a vertex of the polygon in the form of a coordinate pair (x, y). We are told that $V[1]$ is the vertex with the least x coordinate and that the vertices $V[1], V[2], \ldots, V[n]$ are ordered counter-clockwise. Assuming that the x coordinates (and the y coordinates) of the vertices are all distinct, do the following.

Give a divide and conquer algorithm to find the vertex with the largest x coordinate in $O(\log n)$ time.

Additional Space