Homework Assignment 2

TTIC 31010/CMSC 37000-1

due on February 20, 11:59pm
no extensions will be given

Please typeset your solutions. Do not discuss the problems with other students. Do not look up solutions online. Submit your work through Canvas.

Problem 1. In a distant country, a wide river crosses the country east to west. There are m cities $u_1, \ldots, u_m$ on the north bank of the river, listed from west to east (u_1 is the westmost city and u_m is the eastmost city) and n cities $v_1, \ldots, v_n$ on its south bank, also listed from west to east (v_1 is the westmost city and v_n is the eastmost city).

$u_1 \quad u_2 \quad u_3 \quad \cdots \quad u_m$

river

$v_1 \quad v_2 \quad v_3 \quad \cdots \quad v_n$

We are given a set E of pairs (u_i, v_j) of cities that the government wants to connect with bridges. However, the bridges cannot cross each other (a city, however, can be connected with several other cities). The goal is to connect as many pairs of cities from E as possible (a pair (u_i, v_j) of cities is considered to be connected if there is a bridge between them).

Design an efficient dynamic-programming algorithm for this problem, prove its correctness, and find its running time. Do not forget to explain how you initialize the DP-table.

Problem 2. A squirrel would like to eat nuts that grow on a tree, but unfortunately it is tired and cannot travel far. It would still like to eat as many nuts as possible under these circumstances. To model this, consider a tree graph T rooted at a vertex r. We know

- tree T;
- an integer number $n_v \geq 0$ for every leaf v of T; n_v is the number of nuts at u;
- a positive real number d_e for every edge e; d_e is the length of e;
- and a positive real number D; D is the most distance the squirrel can travel.

Initially, the squirrel is located at the root of the tree. Design an algorithm that finds a path for the squirrel that starts and ends at the root, has length at most D, and maximizes the number of nuts collected.

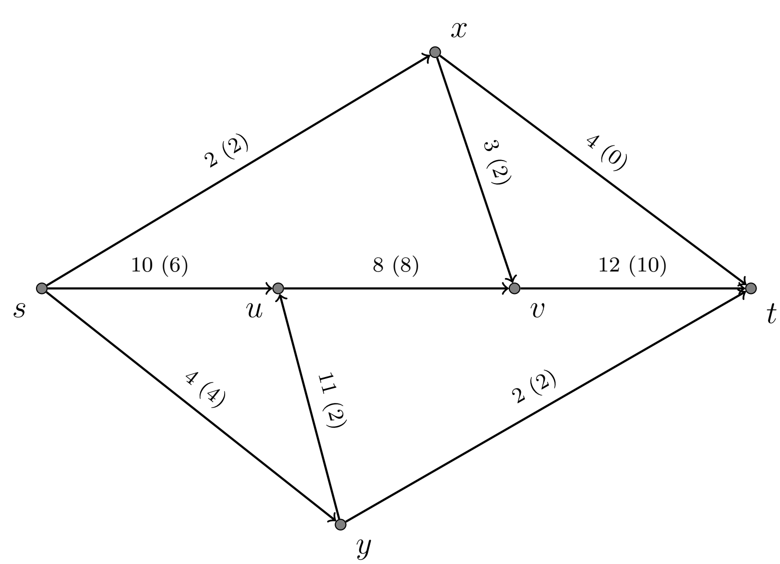
The running time of the algorithm should be polynomial in nM, where n is the number of vertices of T and M is the total number of nuts on the tree. Prove the algorithm's correctness and find its running time. (Partial credit will be given if you solve this problem for a special case where every inner vertex of T has at most two children.)

Problem 3. Let $G = (V, E)$ be a flow network with a source s, sink t, and positive integer edge capacities $c(e)$. Decide whether each of the following statements is true or false. If a statement is true, give a proof; if it is false, provide a counterexample.

- If all capacities $c(e)$ are even, then the value of the maximum flow is even.
- If all capacities $c(e)$ are odd, then the value of the maximum flow is odd.
- If f is a maximum s-t flow in G, then f saturates every edge in $out(s)$. That is, for each $e \in out(s)$, $f(e) = c(e)$.

Problem 4. Consider the flow network G and flow f shown in the figure below. For every edge e, its capacity $c(e)$ and the flow amount $f(e)$ are written next to e ($f(e)$ appears in parentheses).

- Draw the residual graph G_f (draw all forward and backward edges of G_f).
- Write the residual capacity of every edge e of G_f (you can write it next to the drawing of the edge).



Feel free to draw/write your solution for this problem on paper and then scan it.

Problem 5. *Programming assignment. See a separate post for implementation and submission requirements.*

We are given a binary tree $T = (V, E)$ on the set of integers from 0 to $n - 1$. The tree is rooted in 0; each vertex i has child $2i + 1$, if $2i + 1 \in V$, and child $2i + 2$, if $2i + 2 \in V$. That is,

$$V = \{0, \ldots, n - 1\} \qquad E = \{(i, j) : i \in V, j \in V, j = 2i + 1 \text{ or } j = 2i + 2\}.$$

Additionally, we are given a number $q[i] \in \{-1, 0, 1\}$ for every $i \in V$. The charge $q(S)$ of a subtree S of T is $q(S) = \sum_{u \in S} q[u]$. The goal is to partition T into subtrees $T_1, \ldots, T_k$ (k is not fixed) so as to maximize $\sum_{i=1}^{k} q(T_i)^2$.

Design and implement a dynamic-programming algorithm that given $q[\cdot]$ computes the value of the optimal solution.