CS345– ALGORITHMS-II
NITIN SAXENA

ASSIGNMENT 4

POINTS: 50

DATE GIVEN: 23-OCT-2019 DUE: 11-NOV-2019

Rules:

- You are strongly encouraged to work *independently*. That is the best way to understand & master the subject.
- Write the solutions on your own and honorably *acknowledge* the sources if any. `http://cse.iitk.ac.in/pages/AntiCheatingPolicy.html`
- Submit your solutions, before time, to your specific TA. The dynamic student-to-TA mapping would be announced by Pranav.
  Pranav Bisht `pbisht@cse`
  Ashish Dwivedi `ashish@cse`
  Abhibhav Garg `abhibhav@cse`
  Nikhil Shagrithaya `nshagri@cse`
  Abhimanyu `abhimanu@iitk` .
  Preferably, give the TA a printed copy of your LaTeXed or Word processed solution sheet.
- There will be a penalty if you write unnecessary or unrelated details in your solution. (Remember that in the exams you'll get limited space.)
- Clearly express the fundamental idea of your algorithm; sketch the main steps of the pseudocode; give a proof of correctness and that of the claimed time complexity. This decides the distribution of partial marks.
- Problems marked '0 points' are for practice.

Question 1: *[5 points]* Give a quadratic-time algorithm to find the longest monotonically increasing subsequence of a sequence of n numbers.

Question 2: *[8 points]* You are given a convex polygon, having n points P, in the Euclidean plane. A weight function w is also provided such that each triple (i, j, k) of points have an associated weight $w(i, j, k)$.

We want to find a *triangulation* of the polygon so that the sum of the weight of the $(n-2)$ triangles is minimized.

Give a cubic-time algorithm to find the *optimal triangulation*.

Question 3: *[10 points]* Let $G = (V, E, w)$ be a weighted directed graph given via adjacency lists in the input. Give an efficient algorithm to find the length of a negative-weight cycle, in G, that uses the minimum number of edges.

Question 4: *[11 points]* An undirected graph $G = (V, E)$ is called *k-edge-connected* if k is the minimum number of edges that need to be deleted to make G disconnected. Eg. a tree is 1-edge-connected, while a cycle is 2-edge-connected.

Give an $O(|E| \cdot |V|^2)$-time algorithm to compute k.
[*Hint:* Use multiple instances of max-flow.]

Question 5: *[16 points]* Let $G = (L, R, E)$ be an undirected *bipartite* graph with vertex V partitioned as $L \sqcup R$ (i.e. edges E connect vertices in L to those in R). For every subset $A \subseteq L$, the *neighborhood* $N(A)$ is defined as the set of vertices in R adjacent to those in A.

Show that G has a *perfect matching* iff $\forall A$, $|N(A)| \geq |A|$.

Question 6: *[0 points]* Implement longest common subsequence algorithm so that it requires only *linear* space.

Question 7: *[0 points]* Give an $O(n \log n)$ time algorithm to find the longest monotonically increasing subsequence of a sequence of n numbers.

Question 8: *[0 points]* Show that the *longest-path* problem in an undirected unweighted graph reduces to finding a shortest-path in a directed weighted graph (with possibly a negative cycle).

Show that the *Hamiltonian path* problem reduces to finding a longest-path in an undirected unweighted graph.

Thus, finding shortest-paths in general graphs is NP-hard.

Question 9: *[0 points]* Show that network flow maximization problem for *multiple-source multiple-sink* reduces to the one on single-source single-sink.

Question 10: *[0 points]* Consider another variant of network flow problem where each edge has associated *flow lower bound* and *flow upper bound*. How do you find max-flow satisfying these constraints?

□□□