CS 331: Algorithms and Complexity (Spring 2026)
Unique numbers: 53150, 53155, 53160, 53165, 53170, 53175

Assignment 3 - Solution

Problem 1: Short Answer Section

(10 pts) True or false. If true, briefly justify, otherwise, provide a counterexample. When justifying, restrict answers to no more than a few sentences.

1. **(1 pt)** A greedy algorithm follows the heuristic of making a locally optimal choice at each stage, with the hope of finding a global optimum.

2. **(2 pts)** In an exchange proof, first, we define the greedy solution $G = \{g_1, g_2, ...g_j\}$ and optimal solution $O = \{o_1, o_2, ...o_k\}$. Second, we describe where G differs from O. This could mean elements that appear in different orders in each solution, or elements that appear in one solution but not the other, etc. Third, perform an exchange to transform the solution O into solution G, proving that the quality will not decrease by doing so.

3. **(2 pts)** If all edges in an undirected connected graph have the same edge-weight values k, k can be either positive or negative, you can use either BFS or Dijkstra's algorithm to find the shortest path from s to any other node t.

4. **(2 pts)** In the interval scheduling problem, the following greedy template always obtain an optimal solution: Choose the interval x that has the largest time span. If the number of conflicting intervals with it ≥ 2, discard x; otherwise keep x and eliminate all conflicting intervals.

5. **(3 pts)** Suppose that in an arbitrary graph, G, the shortest path from a node, s, to another node, t, is T. If each edge in G is incremented by 1 to obtain a new graph G'. Is the shortest path from s to t in G' always the same as that of G?

Problem 2

(10 points) As a programmer in a startup company, you are assigned a list of independent tasks, numbered $1, 2, \ldots, n$. Completing task i gets you $p(i)$ penalty points. Each task i needs a duration of $t(i)$ to finish and the time at which the task is finished by the programmer is denoted by $c(i)$. Describe a greedy algorithm for identifying a schedule that minimizes

$\sum_{i=1}^{n} p(i) \cdot c(i)$.

You are give the following greedy templates to choose from:

1. (Smallest duration first) Pick task i that has the minimum duration $t(i)$, or
2. (Most valuable first) Pick task i that has maximum $p(i)$, or
3. (Maximum time-scaled value first) Pick task i that has maximum $p(i)/t(i)$.

Choose one of these greedy choices that provides an optimal solution and prove its correctness. Note that only one out of these 3 choices will provide the optimal solution. For the other choices, you only need to provide a counter example.

Problem 3

(10 points) Dijkstra's algorithm was applied to a given directed and connected graph $G = (V, E)$ with positive edge weights starting from the node s. The shortest path from s to another node in the graph t was recorded. Then, the graph was modified by doubling the weight of each edge. Let's call the modified graph G'. Dijkstra's algorithm was then executed on G' and the shortest path from s to t was also recorded. Would the shortest path from s to t in G be the same as that of G' (i.e., both will have the same set of nodes). If yes, provide a proof; otherwise, provide a counter example. You may assume that Dijkstra's will break ties deterministically.