Dept. of Computer Science
University of Toronto

Term Test # 1
(Duration: 50 minutes)

CSC 373 H1
Summer 2021

Student Number: |_|_|_|_|_|_|_|_|_|_|_|_|_|

Last (Family) Name(s): _____

First (Given) Name(s): _____

Write the following statement on the first page of the exam:

"In submitting this exam, I confirm that my conduct during this exam adheres to the Code of Behaviour on Academic Matters. I confirm that I did NOT act in such a way that would constitute cheating, misrepresentation, or unfairness, including but not limited to, using unauthorized aids and assistance, personating another person, and committing plagiarism."

On the first page of your answer sheet, please write down your full name and student number. Also write down the above Academic Integrity Statement.

This test consists of 2 questions.

In your answers, you may use without proof any result or theorem covered in lectures, tutorials, homework, tests, or the textbook, as long as you give a clear statement of the result(s)/theorem(s) you are using. You must justify all other facts required for your solutions.

Write up your solutions carefully! In particular, use notation and terminology correctly and explain what you are trying to do — part marks *will* be given for showing that you know the general structure of an answer, even if your solution is incomplete.

If you are unable to answer a question (or part), you will get 10% of the marks for that question (or part) if you leave it blank, and 20% of the marks if you write "I don't know" and nothing else — you will **not** get those marks if your answer contains contradictory statements (such as "I don't know" followed or preceded by parts of a solution that have not been crossed off).

MARKING GUIDE

1: _____/20

2: _____/20

TOTAL: _____/40

Good Luck!

OVER...

Term Test # 1
(Duration: 50 minutes)

Question 1. [20 MARKS]

OnlineGames.com is a website that sells games in various categories. The website has a total of n games $G = \{g_1, g_2, \ldots, g_n\}$ with respective prices $P = \{p_1, p_2, \ldots, p_n\}$, i.e, game g_i has price $\$p_i$ for $1 \leqslant i \leqslant n$. Assume all prices are integers greater than 10. For the month of April, management decided to run a promotion by offering each game for just \$10. You are a subscriber of the website and you have a budget of $\$B$ with $B > 10$. You saw this opportunity and decided to purchase as many games as possible so as to ***maximize value*** while staying within your budget (**value** for game g_i is defined as $p_i - 10$).

Part (a) [5 MARKS]

Design a Greedy algorithm to solve the above problem. Write your algorithm as a pseudocode. Also, include a clear, concise, high-level English description of the main idea behind your algorithm.

Part (b) [2 MARKS]

Compute the complexity of your algorithm. Explain.

Part (c) [8 MARKS]

Prove the correctness of your algorithm.

Part (d) [5 MARKS]

After one month, management realized it was probably a mistake to drop the prices to \$10. So instead they decided to charge $\$p_i/4$ for game g_i, i.e., they decided to give a 75% discount on ***every*** game. Is the greedy algorithm from part (a) still optimal? If yes, briefly explain why. If not, give a counterexample.

Question 2. [20 MARKS]

In June, management came up with a different policy. They decided to have a single price for all games belonging to a given category, but different prices for different categories. Assume the website has $C = \{c_1, \ldots, c_m\}$ categories, every game g belongs to exactly one category, and every category has at least one game. The prices set for the categories are given by $Q = \{q_1, \ldots, q_m\}$. So all games in category c_1 have price $\$q_1$, all games in category c_2 have price $\$q_2$, and so on. Clearly, the value of game g in category c_i is $p_g - q_i$. Assume the prices in Q are integers and are set so that all games have a positive value. You again have a budget of $\$B$ with $B > 0$, and you want to purchase as many games as possible to **maximize value**. However, there is a restriction. You can only purchase at most 1 game from each category. Design a Dynamic Programming Algorithm to solve the above problem.

Part (a) [5 MARKS]

Define the subproblems and describe carefully the recursive structure of the problem in order to justify the correctness of your recurrence.

Part (b) [3 MARKS]

Define an array for memoization. Properly define what each element of the array stands for. Also, write the recurrence from part (a) in terms of the array elements.

Part (c) [5 MARKS]

Write an iterative, bottom-up algorithm (aka pseudocode) to compute the maximum value obtainable using the recurrence defined in part (a) or part (b).

Part (d) [3 MARKS]

Write an algorithm (aka pseudocode) to find an optimal solution, i.e., a selection of games which when purchased gives the maximum possible value.

Part (e) [4 MARKS]

Compute the space and time complexities of your overall algorithm (combined across parts (c) and (d)).

Total Marks = 40

End of Term Test # 1
(Duration: 50 minutes)