CSE 102: Spring 2021

Quiz 6: Lower Bounds and Computational Complexity (25 points)

45 minutes (Quiz) + 10 minutes (uploading) = 55 minutes
Start Time: 5:25pm; Submission Time: 6:20pm

1. **P-NP** (5 points)
   (a) Does the dynamic programming solution to the Coin Changing Problem belong to P? (True or False).
   (b) Provide a very brief explanation to your answer above.
   (c) Does the dynamic programming solution to the Canoe Rental Problem belong to P? (True or False).
   (d) Does Binary Search Algorithm belong to NP? (True or False)
   (e) Describe a decision problem with following characteristics: (i) no polynomial bound algorithm is known, (ii) which is known to be in NP, (iii) but which is not known to be NP-complete. Write down the decision problem as a full sentence (not just a phrase).
   (f) Coin Changing Problem: We have coins of three denominations: 1, 5, and an unknown denomination d. Change is required for 100. EITHER prove that the greedy algorithm will not use any coins of denomination 1 OR provide a counterexample where coins of denomination 1 may be used.

2. (5 points) **Connected Graph: Adversary Argument**

   You are playing adversary to a player who is trying to guess whether the graph you have in mind is connected or not. Your goal is to force the player to ask as many questions as possible.

   Assume that **the graph has 5 vertices** numbered 1 to 5. Assume that the player will be asking questions in the following order: Is there an edge between $(1, 2), (1, 3), (1, 4)(1, 5), (2, 3), \cdots$? and so on. Write down the adversary answers to these questions. [Clearly justify the responses using appropriate diagrams]. What is the minimum number of questions that the adversary will force the player to ask?

3. (5 points) **Vertex Cover Problem**:

   Find the minimum number of vertices (and call this set vertex cover) in an undirected graph that cover all the edges (that is, every edge is incident on at least one vertex in the vertex cover).

   Example: A graph G of 5 vertices $[1, 2, 3, 4, 5]$.
   The graph has 4 edges $\{(1, 2), (1, 3), (1, 4), (2, 3)\}$, where (i, j) means that there is an edge between i and j. This graph has the smallest vertex cover of $[1, 2]$ with 2 vertices. $[1, 3]$ is also a valid solution.

   Consider the following greedy algorithm: Choose the vertex with the highest degree (the one covering the largest number of edges), add it to the cover, remove all its incident edges from the graph, and repeat.

   EITHER provide an outline of a proof that the above greedy algorithm produces an optimal solution OR

   Provide a counterexample to establish that this greedy algorithm does not always work. Your counterexample must have between 5 to 7 vertices. You can produce the solution using a diagram where vertices are numbered. Be sure to write down the (i) greedy solution in the chosen order, and (ii) the optimal solution.

4. (5 points) **Peeking a Bit String:** Let $b = x_1x_2x_3x_4$ be a bit string of length 4, where each $x_i \in \{0, 1\}$.

   Answer the following questions in any order that helps you work out the solution.

   (a) How many minimum peeks are needed to determine whether the bit string contains two consecutive zeroes?

   (b) If you are the player, which is the first bit that you will peek at? [If there are multiple answers, you can just choose one. You do not have to list them all.]

   (c) Present the player strategy as a decision tree (similar to the solution used in Homework 6 problem) to discover the two consecutive zeroes using minimum number of peeks.

5. (5 points) **Monotonically Decreasing Function: Adversary Argument**

   You are playing adversary to a player who is trying to guess when does a monotonically decreasing real-valued function defined on positive integers (that you have in mind) becomes negative for the first time. Your goal is to force the player to ask as many questions as possible.

   Recall that a monotonically decreasing function $f(n)$ satisfies the following property: $f(i) < f(j)$ for $i > j$.

   The player begins by asking the first question: What is the value of f(1)? You, the adversary responds, 256. [Oops, did you make a mistake? You don't have to answer this question.]

   Assuming that the player plans to ask the values of $f(2)$, $f(2^2)$, $\cdots$, $f(2^n)$,

   (a) What is the number of questions you can force the player to ask before a correct answer can be produced? [$f(1)$ is the first question, $f(2)$ is 2 questions, and so on ...]

   (b) Write down the exact answers (or a strategy) that you will use to answer the player's questions.