[Spring 2022] CSE 317: Design and Analysis of Algorithms [All sections]
Homework 2 | Due: April 27, 2022

Instructions: You should submit it handwritten on A4-size papers. Write your name, section, and ID clearly on top of every page as well as the total number of pages. Direct your all queries to the course staff (refer to syllabus for contact information and office hours).

---

1. Divide and conquer is a technique where the input problem is decomposed into subproblems of smaller sizes recursively and solutions of these subproblems are used to construct a solution to the main input problem.

   (a) (10 points) Explain the difference between dynamic programming and divide and conquer technique.

   (b) (20 points) The Fibonacci sequence is a famous mathematical sequence. The first and second Fibonacci numbers are 1 and 1. The n-th Fibonacci number is defined as the sum of the previous two Fibonacci numbers, where $n > 2$. Design an $O(n)$ dynamic programming algorithm to compute the n-th-Fibonacci number.

   (c) (10 points) Show the underlying DAG of subproblems considered during the computation of the fifth Fibonacci number.

2. (20 points) For a sequence $X = \langle x_1, x_2, \ldots, x_n \rangle$, we define the mirror of X as the sequence $Y = \langle x_n, x_{n-1}, \ldots, x_1 \rangle$. We say that a sequence X is *beautiful* if the mirror of X is the same as X. Design an $O(n^2)$ dynamic programming algorithm that finds the minimum number of elements to add to a given sequence X to turn it into a beautiful sequence. [We can add new elements anywhere in the sequence.]

3. (10 points) Design a linear time dynamic programming algorithm to find the longest path in a directed acyclic graph.

4. (20 points) Suppose you roll a fair k-sided die with the numbers 1 through k on it. If X is the number that appears, what is $E[X]$?

5. (30 points) Let $L = \langle x_1, x_2, \ldots, x_n \rangle$ be a sequence of elements that contains exactly k occurrences of the element x $(1 \leq k \leq n)$. We want to find one j such that $x_j = x$. Consider the following procedure until x is found. Generate a random number i between 1 and n and check whether $x_i = x$. Which method is faster, on the average, this method or linear search? Explain.

6. (30 points) Analyze the following algorithm for its expected running time

   Algorithm: RANDOMSAMPLING

   Input: Two positive integers m and n, s.t. $m < n$.

   Output: An array $A[1..m]$ of m distinct positive integers selected randomly from the set $\{1, 2, \ldots, n\}$.

```
$S[1..n]$ = [false, ..., false]        // $S$ is Boolean array initialized with false values
$A[1..m] = [0, \ldots, 0]$                      // $A$ is an array initialized with 0's
$k \leftarrow 0$
while $k < m$
  $r \leftarrow random(1, n)$       // $random(a, b)$ returns a random number in the range $[a, b]$
  if $S[r] \neq$ True
    $k \leftarrow k + 1$
    $A[k] \leftarrow r$
    $S[r] \leftarrow$ True
return $A$
```
