CSE 312 Butunleme

Olcay Taner YILDIZ

I. Question (Algorithm Analysis) (15 points)

Given the following function

```
int algorithm(int N){
  int i, j, k, sum = 0;
  for (i = 1; i <= N; i++)
     for (j = 1; j <= N; j++)
        for (k = 1; k <= j + i; k++)
          sum++;
  return sum;
}
```

What does the function algorithm return in terms of N?

II. Question (Brute Force) (15 points)

Let A = $\{a_1, a_2, \ldots, a_n\}$ and B = $\{b_1, b_2, \ldots, b_n\}$ be two sets of numbers. Consider the problem of finding their unions, i.e., the set C of all the numbers that are in A or B. Design a brute-force algorithm for solving this problem in $\mathcal{O}(n^2)$ time.

III. Question (Divide and Conquer) (15 points)

Use Divide and Conquer strategy to solve the above problem in $\mathcal{O}(n \log n)$ time.

IV. Question (Graph Algorithms) (15 points)

A graph is said to be in *star* shape if there is only one internal node which is only connected to other N - 1 nodes. Given the adjacency list representation of unweighted, undirected graph G, write a function that returns true if the graph is star shaped or false otherwise.

```
boolean isStar()
```

V. Question (Greedy) (20 points)

Given set of n cities, find the shortest hamiltonian cycle that starts from a city c, visits every city exactly once and returns to the city c. Assuming that there is an edge between every pair of cities and the distances between cities i and j are given by the matrix D_{ij}, return the indexes of cities in the order they are visited.

```
int[] hamiltoniancycle(int[][] D)
```

VI. Question (Dynamic Programming) (20 points)

You are given an $n \times n$ chess board and a pawn. The pawn can move from the bottom of the board to the top of the board according to the following rules:

- The pawn can move one square directly above.
- The pawn can move one square left and one square up (northwest direction).
- The pawn can move one square right and one square up (northeast direction).

If the pawn moves to the square (x, y), it earns $p(x, y)$ points. Given a two dimensional array p, your aim is to find the best route for a pawn starting from the bottom row (in any square) and finishing on the top row (in any square) of the chess board.

```
int route(int[][] p)
```