CSE 531 C: Algorithm Analysis and Design Fall 2023

Programming Project 2

Instructor: Kelin Luo **Deadline: Dec/04/2023**

Your Name: _____ Your Student ID: _____

Problems	1	2	3	Total
Max. Score	10	10	10	30
Your Score				

Requirement:

- You should submit **each source file for each problem**. Name your file: Problem-number_Lastname_Firstname_YourStudents_ID Number. Example: 1_Doe_John_55552222
- Your code need to read the input from the console and output directly to the console.
- Allowed programming languages: Python. Use Python version ≥ 3.4. You can run "python –version" to check its version. Do not use any third-party packages other than the standard library.
- The Programming project 2 deadline is 04 Dec, 11:59PM EST. No late submission is allowed.
- The auto similarity-check will be applied for the programming submissions.
- Only the most recent submission is kept on Brightspace. Note that in each submission, please submit **all** your source files. Only files with extension .py will be accepted for the submission.
- The grading scheme will be released when the programming project 2 grades are published (Tentative release date: 15 Dec).

Problem 1 (10 points). Background: You are given two DNA sequences, sequence1 with size n characters and sequence2 with m characters. DNA sequences are composed of the characters 'A', 'C', 'G', and 'T'. Each DNA sequence may contain two consecutive sequences: sequence1=sequence11+sequence12 and sequence2=sequence21+sequence22. For example, sequence1='AGACTACCG', sequence1 may contain two consecutive sequences sequence11='' and sequence12='AGACTACCG', or sequence1 may contain two sequences sequence11='A' and sequence12='GACTACCG', ... or sequence1 may contain two sequences sequence11='AGACTACCG' and sequence12=''.

Your task is to find two sequences, sequence11 and sequence21, such that: (1) The length of the longest common subsequence (LCS) of sequence11 and sequence21 is equal to the length of the LCS of sequence1 and sequence2. (2) The total number of characters

in sequence11 and sequence21 is minimized. The output should be the length of the LCS and the total length of the two sequences, sequence11 and sequence21, that you have found.

An $O(nm)$-time algorithm is suffice to pass any feasible test cases.

Hint: use the dynamic programming algorithm for longest common subsequence problem.

(a) **Input** You need to read the input from the console. It contains two lines, each containing one sequence. In the first line of the input, we have n characters; and in the second line of the input, we have m characters. You can assume that $1 \leq n \leq 10000$ and $1 \leq m \leq 10000$.

(b) **Output** You need to output to the console. The first line is an integer indicating the length of the LCS between the given two sequences. The second line is the total length of the two sequences, sequence11 and sequence21.

Below are example for input and output:

Example input:	Example output:
AGACTTAA ATGACTATTCA	7 18

Problem 2 (10 points). Given are an undirected graph with n nodes (vertices) and m edges, each having weights. Your task is to find a minimum spanning forest containing no more than k trees. Note that $1 \leq k \leq n$.

An $O(m \log^2 n)$-time algorithm is sufficient to pass all test cases.

Hint: use Kruskal's algorithm.

(a) **Input** You need to read the input from the console. In the first line of the input, we have three positive integers n, m and k. n is the number of vertices in the graph, m is the number of edges in the graph and k is a positive number. The vertices are indexed from 1 to n. You can assume that $1 \leq n \leq 1000$, $1 \leq m \leq 100000$ and $1 \leq k \leq n$. In the next m lines, each line contains 3 integers: u, v and w, with $1 \leq u < v \leq n$ and $1 \leq w \leq 100000$. This indicates that there is an edge (u, v) of weight $w > 0$. You can also assume that the graph is connected.

(b) **Output** You need to output to the console. The output is an integer indicating the total weight of the minimum spanning forest that contains no more than k trees, i.e., the minimum spanning forest that contains $\leq k$ trees.

Below are example for input and output:

Example input:	Example output:
5 5 2 1 2 2 1 3 1 2 3 2 2 4 3 4 5 4	6

Problem 3 (10 points). Given are a directed graph with n nodes (vertices) and m edges, each having weights. Your task is to find the shortest path from vertex 1 to vertex n.

An $O(n^3)$-time algorithm is sufficient to pass all test cases.

(a) **Input** You need to read the input from the console. In the first line of the input, we have two positive integers n and m. n is the number of vertices in the graph and m is the number of edges in the graph. The vertices are indexed from 1 to n. You can assume that $1 \leq n \leq 1000$ and $1 \leq m \leq 100000$. In the next m lines, each line contains 3 integers: u, v and w, with $1 \leq u \leq n$, $1 \leq v \leq n$ and $-100000 \leq w \leq 100000$. This indicates that there is an edge (u, v) from vertex u to vertex v. You can also assume that there is no negative cycle in the graph.

(b) **Output** You need to output to the console. The output is an integer indicating the total length of the shortest path from vertex 1 and vertex n. If there is no path from vertex 1 to vertex n in the graph, the output is INFINITY.

Below are example for input and output:

Example input:	Example output:
5 5 1 2 2 1 3 -1 2 3 2 2 4 3 4 5 4	9

Example input:	Example output:
5 4 1 2 2 1 3 -1 2 3 2 5 4 1	INFINITY