Please note that there are two problems (Problems 1 and 2) under this assignment. See below for Problem 2.

Problem 1 (10 Marks)

Longest Common Subsequence

Implement the dynamic programming solution to the longest common subsequence problem (LCS). (Reference: Section 15.4 of Cormen's Book). Refer to this section of your book to find the definition of subsequences and common subsequences.

Given two sequences $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_n)$ find a maximum length common subsequence of ***X*** and ***Y***. You have to output both the subsequence length and the subsequence itself.

Input will consist of the two strings, they would be of at most of length 50 and will only consist of capital letters 'A' to 'Z'. The first line of output will consist of the length of the longest common subsequence. The second line will contain the LCS itself. If there are more than one LCS, you may print any one of them.

Sample Input	Sample Output
ABCBDAB BDCABA	4 BCBA
ACCGGTCGAGTGCGCGGAAGCCGGCCGAA GTCGTTCGGAATGCCGTTGCTCTGTAAA	20 GTCGTCGGAAGCCGGCCGAA

Problem 2 (10 Marks)

Optimal Cost to Visit Cities

Bob went to visit a place called Wonderland, which has ***n*** cities. He wants to visit all ***n*** cities before leaving Wonderland. Now the government of Wonderland enforces some weird rules for visiting tourists. Upon entering Wonderland, tourists have to pass a checkpoint, where they are handed a log file. They have to submit this file when they leave Wonderland.

Initially the log file is empty. Whenever the visitor enters a city to visit, government officers check the log file and add the city's name to the log file. Now, the visit isn't free of charge. Each city ***i*** has a certain cost C_i which has to be paid before entering the city. In addition to this, if the officers of the city ***i*** see a visitor has already visited city ***j*** in Wonderland (as shown in the log file), they will also additionally charge a cost $A_{i,j}$. Thus upon entering city ***i***, a visitor has to pay: $C_i + \sum A_{i,j}$, for all cities ***j*** already visited by the visitor.

Now, Bob wants to know the minimum amount of money he needs to visit all ***n*** cities. Help Bob to find this minimum cost.

The first line of input contains an integer ***n (n <= 10)***, the number of cities in Wonderland. The next ***n*** lines will contain ***n*** integers (the value of all integers being between 0 to 10000) each. Assume this is an ***n x n*** matrix ***A***. ***A[i,i]*** denotes the cost C_i to visit city ***i***, while ***A[i,j]*** denotes the extra cost a visitor has to pay at city ***i*** if he has already visited city ***j*** before. Output the minimum amount of money that is required to visit all ***n*** cities.

Sample Input	Sample Output
2 100 100 5000 100	300
3 25 100 0 0 25 0 150 100 25	75
3 30 75 95 120 45 20 105 30 90	355

Hint:
You may use a technique called bitmasking in dynamic programming (the problem 2 above is a bitmask+DP type problem). You can find some examples of bitmask in dynamic programming in the following links:

https://www.geeksforgeeks.org/bitmasking-and-dynamic-programming-set-1-count-ways-to-assign-unique-cap-to-every-person/?ref=lbp

https://www.geeksforgeeks.org/bitmasking-dynamic-programming-set-2-tsp/?ref=lbp

Special Instructions

Write ***readable, re-usable, well-structured, quality*** code. This includes but is not limited to writing appropriate functions for implementation of the required algorithms, meaningful naming of the variables, suitable comments where required, proper indentation etc.

Please DO NOT COPY solutions from anywhere (your friends, seniors, internet etc.). Implement the algorithms with your style of coding. Any form of plagiarism (irrespective of source or destination), will result in getting -100% marks. You have to protect your code.

Also, be informed that for the repeated offense of plagiarism, the departmental policies suggest stricter measures.

Submission Guideline

1. Create a directory with your 7 digit student id as its name
2. Put the source files only into the directory created in 1
3. Zip the directory
4. Upload the zip into moodle

For example, if your student id is 1905123, create a directory named 1905123. Put your source files(.c, .cpp, .java, .py, .h, .hpp etc) only into 1905123. Zip 1905123 into 1905123.zip and upload the 1905123.zip into moodle.

Failure to follow the above-mentioned submission guideline will result in up to 10% penalty.

Submission Deadline:

Feb 12, 2021 10:50 AM

This is a hard deadline and there shall be no extensions for any reason whatsoever.