
CS29003 ALGORITHMS LABORATORY
ASSIGNMENT 5 (Dynamic Programming)
Date: 8th October, 2020

Important Instructions

1. **List of Input Files:** input.txt
2. **Output Files to be submitted:** ROLLNO_GroupNO_A5.c/.cpp
3. Files must be read using file handling APIs of C/C++. Reading through input redirection isn't allowed.
4. You are to **stick to the file input output formats strictly** as per the instructions.
5. Submission through **.zip files are not allowed**.
6. Write your name and roll number at the beginning of your program.
7. Do not use any global variable unless you are explicitly instructed so.
8. Use proper indentation in your code.
9. Please follow all the guidelines. Failing to do so will cause you to lose marks.
10. **There will be part marking.**

Problem Statement

Recently you have joined a chemistry lab and they are leading a research on a system which consists of a sequence of independent and essential chemical reactions. Surprisingly the researchers of the lab have discovered a unique catalyst and it can be included in any of the reactions of the system. The use of the catalyst enhances the rate of the reaction and hence its probability of success also. Also they have proven that the probability of success of the reactions depends on the usage of the catalyst, typically expressed in terms of number of units involved in the reaction. The more the quantity (units) of catalyst is used in a reaction, the probability of success increases or remains the same, but never decreases. The probability of success of the entire system is the product of the probability of success of each of the individual reactions. Given the limited quantity (units) of the catalyst, the researchers want to gain the maximum success of the system. As an expert in algorithms, you are given the task of maximizing the success probability of the system with the constrained amount of the catalyst. You have to formulate a **Dynamic Programming solution** for the problem (use of memoization is **NOT** allowed).

You are given N reactions, numbered from 1 to N , all of which must be completed for the successful completion of reactions of the system. The system has a total of C units of catalyst ($C \geq N$) that can be used in different reactions. Let $e(k, p)$ denote the probability of success of the reaction k if p units of catalyst are used in the reaction. It is given that $e(k, 0) = 0$ for all k , so at least one unit of catalyst must be used by each of the reaction. Also, for a reaction k , $e(k, p)$ values are non-decreasing with increasing values of p (i.e., the chance of success never decreases if more units of catalyst are assigned to the reactions). You need to assign the C units of the catalyst to the N reactions so that the probability of success of the system is maximized.

You are required to design an efficient dynamic programming algorithm for the problem.

Part I

Write as comments at the very beginning in your program, the following clearly

1. The definition of the subproblem that you choose.
2. The recursive formulation used in the DP solution, including the base case. Write using notations as you have seen in text, do not write long sentences.

Partial marks will be awarded if you can figure out the recurrence.

Part II

Write a `main()` function to implement the dynamic programming. Read the input file as follows

1. The first line contains the number of reactions N (assume $N < 10$)
2. Read in C (assume $C < 30$. You can also assume C will be entered as $\geq N$, no need to check).
3. Read in the $e(k, p)$ values for each reaction exactly in this order in a 2-d array (enter the values in each row in non-decreasing order):
 $e(1, 1), e(1, 2), \dots, e(1, C)$
 $e(2, 1), e(2, 2), \dots, e(2, C)$
 \dots
 $e(N, 1), e(N, 2), \dots, e(N, C)$

Sample input file

FILE: *input.txt* _____

```
3
5
0.1 0.3 0.5 0.6 0.6
0.1 0.2 0.4 0.6 0.8
0.2 0.4 0.4 0.7 0.9
```

You have to print the followings exactly same as shown in the sample output file

1. The maximum probability of success, of the system.
2. The assignment of the catalyst in units to each of the reaction that gives the maximum success probability of the system.

Sample output file

FILE: *output.txt* _____

```
0.012
reaction 1 : 2
reaction 2 : 2
reaction 3 : 1
```

Notes

1. There may be more than one assignment that gives the same maximum success probability; in such cases, you can print any one assignment that gives the maximum success probability.
2. You have to code directly in *main()*, no separate function is allowed to use.
3. You are not allowed to use **Memoization**
4. Make sure to read inputs from a file named "*input.txt*". At the time of evaluation, the input data might be different from the one given here.
5. You need not to submit "*output.txt*", but your code should write the output in the given format in a file named "*output.txt*"