ShanghaiTech University

CS240 Algorithm Design and Analysis

Spring 2022

Course Project

Due: 23:59, June 19, 2022

1. This project requires you to solve four problems. For each problem, write a program with the input and output formats as specified in the problem.
2. All programs must terminate within the limits given in the problem. Programs exceeding either limit will not be accepted. You can write your programs in C, C++, Java or Python, though certain languages may lead to more efficient implementations. Note that the online judge which will run your program can perform around **1e8-1e9** arithmetic calculations per second, so if the time limit is 1 second, your program can not perform more than 1e9 calculations.
3. You may not use third-party libraries in your solution, e.g. numpy in Python is not allowed. If you are not sure whether a library is third-party or not, you can try to submit your program, and it will cause a Runtime Error or Compile Error if it is.
4. Your program submission is scored by an Online Judge platform containing a number of test cases. Your score depends on the percentage of test

cases your program passes.

5. The Online Judge website is http://10.19.126.173.

6. Each student's account name and initial password are both **STUDENT ID**. If you have a **STUDENT ID** with ten digits, please try it first. If you can't log into OJ, please email your **STUDENT ID** to tanhy@shanghaitech.edu.cn. After logging into your account for the first time, please change your password.

7. **DO NOT** use scripts to submit code or submit the code too frequently. If you do that, we will consider you to be maliciously testing the data and give you a 0 for the problem.

8. You must NOT

   (a) Read or use solutions written by others.

   (b) Allow other people to read or use your solutions.

   (c) Obtain test data by repeated submissions or other means.

Problem 1:

Suppose you are given a string which contains at most three different characters 'A', 'B' and '?'. You need to change all '?' to 'A' or 'B', and will then pay a cost for the resulting string. In particular, for each occurrence of "AB", you pay X, and for each occurrence of "BA" you pay Y. Design an algorithm to minimize the cost you pay.

Input

The first line of the input gives the number of test cases T. Then T lines follow. Each line contains two integers X and Y and the initial string.
Assume the length of the string is ≤ 1000, $-100 \leq X \leq 100$, $-100 \leq Y \leq 100$, $1 \leq T \leq 100$.

Output

For each test case, output one line containing 'Case #x: y', where x is the test case number (starting from 1) and y is the minimum cost that you need to pay.

Limit

Time limit: 1 second
Memory limit: 512MB

Sample Input

5
2 3 AB?AA?
4 2 ABAB
1 3 A?B
2 5 ??B???
2 -5 ??BB??

Sample Output

Case #1: 5
Case #2: 10
Case #3: 1
Case #4: 0
Case #5: -8

Problem 2:

Suppose you are given a graph with positive edge weights, and a start and destination node. You need to design an algorithm to find two paths from the start to the destination node, such that the paths do not share any edges, and the sum of the lengths of the two paths is as short as possible.

Input

The input will contain several test cases. For each test case, the first line will contain an integer n ($2 \leq n \leq 100$) which is the number of nodes in the graph. The start node is number 1, and the destination node is number n. The next line will contain an integer m which the number of edges. The next m lines will describe the m edges. Each line will contain 3 integers, which are the two endpoints of the edge and the edge's length. No edge will be longer than 1000 or shorter than 1. Each edge connects two different nodes, and no pair of nodes will be directly connected by more than one edge. The last test case will be followed by a line containing zero.

Output

For each test case, if there is a solution, output a number on a line by itself, which is the sum of the lengths of the two paths. If there is no solution, print "Back to jail".

Limit

Time limit: 3 seconds
Memory limit: 512MB

Sample Input

```
2
1
1 2 999
3
3
1 3 10
2 1 20
3 2 50
9
12
```

1 2 10
1 3 10
1 4 10
2 5 10
3 5 10
4 5 10
5 7 10
6 7 10
7 8 10
6 9 10
7 9 10
8 9 10
0

Sample Output

Back to jail
80
Back to jail

Problem 3:

You are given an $R \times C$ matrix containing unknown values. You are told the sums of the values in each row and each column. Find the values in the matrix.

Input

The first line of the input is the number of test cases, which is between 1 to 100. Each test case contains 3 lines of input. The first line of the test case contains R ($1 \leq R \leq 20$) and C ($1 \leq C \leq 20$), which are the number of rows and columns. The next line contains R numbers, where the i'th number is the sum of all the values in rows 1 to i. The last line contains C numbers, where the i'th number is the sum of all the values in columns 1 to i.

Output

For each test case, first print "Matrix x" on a separate line, where x is the test case number. Then print an $R \times C$ matrix which satisfies the condition of the input. Separate each number with a space, and output each row on a separate line. Separate output matrix with a blank line. Each test case is guaranteed to have a solution. To simplify the problem, we add the constraint that each entry in the matrix must be an integer between 1 and 20. If there are multiple solutions, you can output any one of them.

Limit

Time limit: 3 seconds
Memory limit: 512MB

Sample Input

```
2
3 4
10 31 58
10 20 37 58
3 4
10 31 58
10 20 37 58
```

Sample Output

```
Matrix 1
1 6 1 2
```

1 2 2 16
8 2 14 3

Matrix 2
1 1 1 7
1 1 7 12
8 8 9 2

Hint: **Do not** use linear equations to solve this problem. It will **time out**.
Note: Note that in the first sample input, the row sums are 10, 31, 58, indicating that the first row sums to 10, the first two rows sum to 31, and all three rows sum to 58. The other row and column sums are computed in the same way.

Problem 4:

You are planning to pay your employees using rare coins. The coins come in n different denominations, where the value of each denomination is a multiple of the previous one (e.g. 1, 5, 10, 20, ... RMB) if they are sorted in an increasing order. You have a limited number of coins of each denomination. You need to pay each employee at least C RMB each month (you can pay an employee more than C, but not less). What is the maximum number of months which you can pay your employees?

Input
The first line contains 2 numbers, n and C, where $1 \leq n \leq 20$ is the number of different denominations, and $C < 100,000,000$ is the amount you need to pay each employee every month. The next n lines each contain 2 number V and B, where $V < 100,000,000$ is the value of a denomination, and $B < 10,000,000$ is how many coins you have of that denomination.

Output
The maximum number of months you can pay your employees.

Limit
Time limit: 1 seconds
Memory limit: 512MB

Sample Input
3 6
10 1
1 100
5 120

Sample Output
111