Project of COMP 6651

Project due date: April 7, 11:55 pm

The Covid-19 situation is an unprecedented and unfortunate event. I hope you and your family all in good health and taking care of yourself. Due to this circumstances the project of this course has changed.

For the project you need to implement solutions for three problems. Each question have a time and memory limit, e.g. 2(s) and 1Mb. The rules and grading scheme are the following:

Estimating how long an algorithm takes or how much memory it uses is one of the aspects of this project. Usually execution of 10^9 unary operations is considered to take one second.[1] This is one of the places that the use of asymptotic complexity (with consideration of few constant factors) is very helpful.

Although there is a judge program for the questions, the marking of this project will not depend on it, as I cannot make the judge available online and that many cases are nuanced (e.g. accuracy consideration, special cases, etc.) and it is not the purpose of this course to pay attention to these, albeit that they are important.

The accepted languages for this project are Java, C/C++, and Python. You may use some libraries for simple tasks such as min, max, sort, etc., but you may not use them for more complicated task such as applying shortest paths algorithm, instead you should implement them yourself.

For each question, a few sample test cases are provided. If your program fails any of them you will not be given any point for that question. Be sure to respect the input and output format so that the judge program has no problem. If you pass all the cases of the judge you will get a bonus mark.

You should submit one PDF documentation and one file containing code for each question, the total of 3 files and 1 PDF. In the documentation you should provide screenshots that proves the sample tests are correctly executed, and also give a short argument explaining why the time and memory limits are respected (Estimate the number of unary operation asymptotically and only consider the most impactful coefficient, also consider the `int` type to take 32bits of space.) If you cannot design an algorithm that respects the limits, provide a less efficient algorithm and argue what its limitation is for a possible partial mark.

Please send a `.zip` file containing exactly four files, with the following name convention:

1. `<net name>-<student id>-doc.pdf`
2. `<net name>-<student id>-<question number>.<file format`[2]`>`

Thank you and stay safe.

1 Socrates' Cows

- Time Limit: 2 seconds
- Memory Limit: 256 Megabytes

[1] The time depends on many factors, for example the architecture and clock speed of the CPU, or even the programming language that is used. In competitive programming problems usually only the clock speed of the CPU is considered. The clock speed is expressed in terms of GHz which is number of cycles in one second. Since the targeted algorithms are not (usually) concurrent, one can estimate the time by estimating the number of unary operations (clocks). The approximate CPU's clock speed for personal computers are usually around 4GHz. For this reason (approximating 4 clocks for each unary operation) we approximate 10^9 unary operations to take 1(s). Also note that the time limits are usually not very tight, and mostly are asymptotically tight.

[2] Depending on the programming language used, e.g. .cpp.

Socrates' cows are grazing in meadows. At each sunset Socrates blows a special whistle and the cows come back to the stable. Socrates wants to know which cow(s) reaches the stable the fastest.

Each cow is in some meadow and some meadows might not have any cows. There are paths between some pair of meadows (it is possible that two meadows have more than one path connecting them.) Starting from any meadow a cow can reach the meadow in which the stable resides. Socrates' cows are smart and they always choose the shortest path to the stable.

All cows move with a same constant speed, and multiple cows can use a same path at the same time. The meadows names are a small or capital English letter, i.e. `a ... z` and `A ... Z`. Before the whistle blows, a meadow has a capital letter name if and only if a cow reside in it. The stable is in meadow `z` and there is no cow in it before the whistle blows.

Input

- Line 1: Integer number P. (Number of paths that connect the meadows.)
- Lines 2 to P+1: Two letters and an integer number in each line, representing a path and the time it takes the cows to pass it.

$$1 \leq P \leq 10^4 \qquad 1 \leq \text{distance} \leq 10^3$$

Output

One line containing one letter and one integer number, representing any meadow whose cow(s) reaches the stable the fastest (therefore definitely a capital letter) and the amount of time it take these cow(s) to reach the stable.

Sample Test Cases

- Input

```
5
A d 9
B d 3
C e 9
d z 8
e z 3
```

- Output

```
B 11
```

2 Red

- Time Limit: 1.5 seconds
- Memory Limit: 512 Megabytes

Little red riding hood is a very competent graph theorist. She has n intervals $[l_i, r_i]$. She created a n vertex graph where each vertex represents an interval. There is an edge between u and v if and only if their corresponding intervals (call them interval u and v) intersect, i.e. $\max(l_v, l_u) \leq \min(r_v, r_u)$. Now she want you to help her to count the number of *Dominating sets* in this graph.

Set S of vertices of graph G is a Dominating set if and only if each vertex v of G is either a member of S or one of its neighbor is in S.

Input

Line 1: Integer number n, number of intervals.
Lines 2 to n+1: l_i and r_i the start and end point of interval i.

$$1 \leq n \leq 5000 \qquad 0 \leq l_i \leq r_i \leq 10000$$

Output

One integer which is the number of Dominating sets of the graph modulo $10^9 + 7$ (be careful about overflows).

Sample Test Cases

- Input 1

```
2
1 5
3 3
```

- Output 1

```
3
```

- Input 2

```
3
1 3
2 5
4 6
```

- Output 2

```
5
```

3 Mr. Meeseeks' Bakery

- Time Limit: 2 seconds
- Memory Limit: 256 Megabytes

In yet another dangerous adventure Morty joins his scientist grandfather Rick to buy a loaf of bread for breakfast from one of the best bakeries in Faraway land. The residents of Faraway land are kind but impatient creatures; if they get bored, they become merciless and will burn everything within their vicinity.

Mr. Meeseeks, an old friend of Rick and Morty, is the owner of the bakery. He is alone and has a lot of customers. He is thrilled to see Rick and Morty, and he asks them for help. They divide the duties in the following manner, Rick and Mr. Meesseks bake, and Morty will give the loaves of bread to customers.

As usual, there are n different queues in the bakery. Each customer wants one loaf of bread and will leave after buying it. Morty can choose the front person of one queue and sell a loaf of bread to that person. No new customer can enter the bakery.

Mr. Meeseeks and Rick together can bake a loaf of bread in a second. Customer j of queue i has the patience level of $p_{i,j}$ (a positive integer number), if this customer does not receive a loaf of bread by this time after they start baking, (s)he will burn the bakery.

If the bakery is on fire, Rick and Morty can escape using their teleport machine. Morty loves this work and wants to do it as long as possible, until all customers are served or one of them set the place on fire. What is the maximum number of customers Morty can sell a loaf of bread to?

Input

Line 1: Integer number n, number of queues.

Lines 2 to n+1: An integer number l_i, the number of customers in queue i. Followed by l_i integer numbers $p_{i,1},\ p_{i,2},\ \ldots, p_{i,l_i}$, the patience level of ith queue customers.

$$1 \leq n \leq \sum_{1}^{n} l_i \leq 100\,000 \qquad 1 \leq p_{i,j} \leq 10^9$$

Output

Maximum number of customers that Morty can sell a loaf of bread to.

Sample Test Cases

- Input 1

```
2
1 1
2 9 2
```

- Output 1

```
2
```

- Input 2

```
3
2 1 2
2 3 5
1 4
```

- Output 2

```
5
```

- Input 3

```
3
1 3
1 4
2 5 2
```

- Output 3

```
4
```