ESO207

Data Structures and Algorithms

Indian Institute of Technology, Kanpur

Date of Submission: Jan 30, 2024

Programming Assignment

1

General Instructions.

- Solution of Programming Assignments(PA) are accepted on Gradescope. Individual submissions via emails will not be graded.
- Gradescope will accept solutions only in the following programming languages - C and C++.
- The autograding process will evaluate your assignment against a predefined set of *visible* test cases immediately upon submission.
- Following the deadline, the final version of your submitted code will undergo autograding with a set of *hidden* test cases.
- Use of algorithm header file like `<algorithm.h>` and primitive functions for operations like sorting, searching and implementing data structures constitutes plagiarism. This course is on Data Structure and Algorithm; hence, we expect students to write their code rather than using predefined function.
- Use of generative tools like ChatGPT is unacceptable and will lead to heavy penalties.

Problem 1: Penalty

Guidelines

- Please submit your singlefile solutions via Gradescope, ensuring that the filename is either `penalty.c` or `penalty.cpp`. The solution should consist of a single file and not multiple files.

Statement

Shrey is participating in the ESO207A Programming Contest. He has been given N questions, each associated with it's penalty value $v_1, v_2, \ldots v_N$ such that $0 \leq v_i \leq 9$ for all

$1 \leq i \leq N$. Shrey has a constraint that he can solve only k questions in the contest. Once the contest is over, he is left with $N - k$ unsolved questions, and the string formed by concatenating the penalty values of the unsolved questions in the same order is converted to an integer, which is considered the net penalty incurred by Shrey in the contest (the final penalty value should not have any preceding zeros). Help Shrey minimize the penalty incurred in the contest!

Input Format

The first line of the input contains the number of test cases. For each test case, the first line contains two integers N and k. And the second line of each test case contains an N digit integer s where the i-th digit represents v_i. For example,

```
2
5 1
10200
7 3
1432219
```

Output Format

The output of each test case must be an $N - k$ digit integer representing the minimum penalty possible for the test case. Suppose, $p_1, p_2, \dots p_{N-k}$ are the unsolved problems such that $p_1 < p_2 < \cdots < p_{N-k}$, then the output is $N - k$ digit number such that the j-th digit is v_{p_j} without any preceding zeros. The output of each test case must be printed in a new line.

```
200
1219
```

Constraints

- $1 \leq k \leq N \leq 10^5$
- $0 \leq v_i \leq 9$

Example

Input

```
1
5 1
10200
```

Output

```
200
```

Explanation: For the first test case, we are given the number 10200 of length 5. Shrey can solve 1 question from these, so if he solves the first the remaining problems $2, 3, 4$, and 5 remain unsolved. Hence the output should be v_2, v_3, v_4, v_5, which is 0200, but the output should not have any preceding zeros, so the output is 200. The number 200 is the minimum penalty possible.

Problem 2: Jump

Guidelines

- Please submit your singlefile solutions via Gradescope, ensuring that the filename is either `jump.c` or `jump.cpp`. The solution should consist of a single file and not multiple files.

Statement

Spider-Man has lost his ability to shoot spider-webs due to an unfortunate accident. He can now only jump between buildings.

You will be given a map of New York City. For every street in the city, you will be given a sequence of heights of the buildings in the street. For every building, Spider-Man can only jump to buildings that are just before the current building in the street-sequence and that are of a lesser or same height as the current building (Note that he can jump on the same building too). If, while looking back at the buildings that came before in the sequence, he encounters a building that is taller than the one he is currently standing, he can't jump past that. Also, he can't jump between streets.

Now, Spider-Man wants to map out all the streets so that he can fight criminals even with his disability. You have to help him figure out for every building in every street, how far back he can jump.

Input Format

The first line contains S, the number of streets in the city. The following lines give details of every street. For every street k, ($1 \leq k \leq S$), you are first given N_k, the number of buildings in the street. The next N_k lines give, in sequence, the heights of the buildings in the street.

Note that all numbers wil be non-negative integers. For example,

```
1
8
10
8
9
5
4
6
7
9
```

Output Format

For each street k ($1 \leq k \leq S$), you have to print N_k lines, where line i ($1 \leq i \leq N_k$) is JB_i, the number of buildings Spider-Man can jump back from building i (including building i).

```
1
1
2
1
1
3
4
7
```

Constraints

- $1 \leq S \leq 1000$
- $1 \leq N_k \leq 1000000$

Example

Input

```
1
8
10
8
9
5
4
6
7
9
```

Output

```
1
1
2
1
1
3
4
7
```

Explanation: There are 8 buildings of heights $10, 8, 9, 5, 4, 6, 7$ and 9 in that order. For building 1, Spider-Man can only jump on the same building :(. From building number 3, he can jump on the same building, but also to building number 2 (height 8). So, $\mathsf{JB}_3 = 2$. And so on. Lastly, from building 8, he can jump to all the previous buildings except for the first one (note that he can jump past building 3 since it's of the same height). So, $\mathsf{JB}_8 = 7$.