

3 Radical Sorting

3.1 Problem description

You have N strings that you want to sort Radically.

String A occurs **before** a string B in Radically sorted order if and only if at the first position of difference from left, the following holds:

- it is an odd position and A has a letter that appears earlier in the alphabet than the corresponding letter in B
- it is an even position and A has a letter that appears later in the alphabet than the corresponding letter in B
- A is a prefix of B and A has an odd length
- B is a prefix of A and B has an even length

For example, consider two strings $cbfz$ and $cefb$. The first point of difference from left is at index 2. As 2 is **even** and e comes **after** b in the alphabet, therefore $cefb$ comes **before** $cbfz$ in a radically sorted order.

Given N strings, sort them Radically.

NOTE: 1-based indexing is followed to determine if the position is odd or even.

NOTE: `<string.h>` is **not allowed** in this question.

3.2 Constraints

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq N \leq 2 \times 10^5$
- Sum of $N * M$ over all test-cases does not exceed 6×10^5
- N is the number of strings
- M is the maximum length of the string
- All the strings contain **only** lowercase english alphabet

3.3 Input format

The first line of input contains a single integer T that denotes the number of test-cases. Then the description for each test-case is as follows.

The first line of each test-case contains a single integer N that denotes the number of strings.

Then N lines follow for each test-case, the i^{th} of which contains an integer m , followed by a string of length m .

That is, it is in the following format

```
T
N
m_1 string_1
m_2 string_2
...
```

3.4 Output format

For each test-case, output the radically sorted strings, each string on a new line.

3.5 Example test-case

Input	Output
2	
3	
3 abc	azcd
4 azcd	abc
4 abcf	abcf
5	abd
3 bdf	bdf
2 zz	baa
1 z	z
3 baa	zz
3 abd	

4 The Schoolyard Shuffle

4.1 Problem Description

K schools meet together every year to host the Schoolyard Shuffle. Each school sends M students. Each student has a unique combat power. The combat powers of $K \cdot M$ students from all schools are given in an array A , arranged in a specific pattern.

Students of School 1: $A[1], A[K + 1], A[2K + 1], \dots, A[(M - 1)K + 1]$
Students of School 2: $A[2], A[K + 2], A[2K + 2], \dots, A[(M - 1)K + 2]$
 \vdots
Students of School K : $A[K], A[2K], A[3K], \dots, A[M \cdot K]$

The first K students fight against each other, and the one with the maximum combat power among them wins. This is followed by the next K students, and so on. Each student can fight only once.

A random school J is chosen each year which is given the special privilege of shuffling the order of it's students in any order at most once. Find the maximum number of wins school J can get at the end of the competition.

NOTE: 1-based indexing is followed in the problem description.

4.2 Constraints

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq M \leq 2 \times 10^5$
- $1 \leq J \leq K$
- $1 \leq A[i] \leq 10^9 \forall i \in [1, K \cdot M]$
- Sum of $K * M$ over all test-cases does not exceed 10^6
- K is the number of participating schools
- M is the number of students from each school
- J is the chosen school that is allowed to shuffle it's students

4.3 Input Format

The first line of input contains a single integer T that denotes the number of test-cases. Then $3T$ lines follow.

The first line of each test-case consists of 2 space separated integers K and M .

The second line of each test-case contains $K \cdot M$ space-separated integers $A[1], A[2], \dots, A[K \cdot M]$ that denotes the combat powers of each participating student. Each combat power is unique.

The third line of each test-case contains an integer J , denoting the chosen school.

That is, it is in the following format

```
T
K M
A_[1] A_[2] ... A_[K*M]
J
```

4.4 Output Format

For each test-case, output a single integer representing the maximum number of wins the school J can get.

4.5 Example Test case

Input	Output
1 3 5 8 5 3 14 9 22 29 13 2 36 47 17 50 51 11 2	3

Explanation:

Here, in the first test-case

Students of School 1: 8, 14, 29, 36, 50
Students of School 2: 9, 51, 5, 47, 13
Students of School 3: 3, 22, 2, 17, 11

is one of the optimal arrangements where the 2nd school has the privilege to shuffle it's students which gives an output 3 by obtaining a win in the first, second and fourth fights.