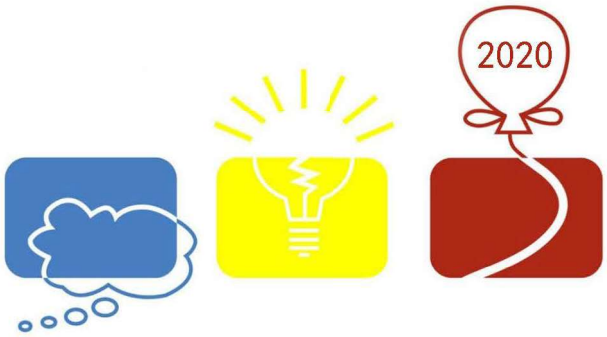
2020 Wuhan University Collegiate Programming Contest

(Preliminary Contest)

April 12th, 2020

武漢大學
WUHAN UNIVERSITY



icpc International Collegiate Programming Contest

Problems

Problem A A Monument for Heroes

Problem B Best Match

Problem C Can You Help ZSGW

Problem D DIY Masks at Home

Problem E E Yu is a Brutal Creature

Problem	A	B	C	D	E
Time Limits	1s	1s	1s	5s	1s
Memory Limits	256M	256M	256M	256M	256M

Problem A. A Monument for Heroes

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Months ago, the COVID-19 swept Wuhan. In order to control the outbreak, many people tried their best to help Wuhan, and with their help, we finally (almost) beat the virus, so we are going to set up a monument and write the names of all the contributors to express our gratitude to them. But there are so many people who have helped Wuhan so there are so many names to write, and how to arrange them is a big problem, so we decided that we will choose some representative names and rearrange their order to make them neatly. But until now we still do not know the maximum length of letters that will be write on the monument. We cannot leave enough space for these names without this number, so we leave this problem for you.

Now you will be given the name of all the contributors, and you can choose some and connect them in a certain order to make them neat. A permutation of names is neat only when for every component except the last one, the last letter of the component is the same as the first letter of the next component, and the last letter of the whole name is the same as the first one. E.g. if there are 3 contributors called "abc", "fga", "cdef", we can choose all these three names and combine them to "abccdeffga" to write on the monument with length of 10.

What's more, these contributors agreed that one who contributed more is more important. They will provide you a preference list according to their agreement and you should connect the permutation **strictly** according to the list. That is, if a name s appears earlier than name t in the preference list, s would also appear earlier than t if they are both selected into the permutations.

There will be many permutations which are neat, but we should leave enough space for every possible one, so can you tell us the length of the longest one?

Note: there might be multiple contributors have the same name.

Input

The first line contains integer n , $(1 \leq n \leq 2 * 10^5)$ — the number of contributors. Next n lines contain n names, one per line. A contributor's name is a non-empty sequence of lowercase Latin letters. Its length does not exceed 10 characters.

Meanwhile, this is a preference list, the name comes earlier should also appear earlier in your final permutation.

Output

Print a single number – the maximum length of combination of names. If there are no possible permutations, print 0 instead.

Examples

standard input	standard output
3 abc ca cba	6
4 aab aab lbwnb mlslj	0

Note

For the first example, the final perutation will be 'abccba'.Please note that 'cbaabc' is against the rule that 'abc' should be important than 'cba'.

Problem B. Best Match

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

There is a traditional activity held by school of computer science in WuHan University: Five-day lovers. But this year, due to COVID-19, this event has to be held online and **HamsterW** wants to organize this activity to help schoolmates who have no girlfriend or boyfriend.

HamsterW knows a lot about opposites attract theory. That is the key principle of this Five-day lovers. The Five-day lovers activity has classified each registered schoolmates according his/her interests and assigned to the $i-th$ schoolmate number $t_i(-20 \leqslant t_i \leqslant 20)$. A number will be assigned randomly to any schoolmates.

HamsterW wants to advertise this activity and inform the number of opposite matches, that is, matches who have opposite values of t. There are two schoolmates in a match. The schoolmate can join multiple groups at the same time.

For example, if $t = (1, -1, 1, -1)$, then any two elements t_i and t_j form a match if $i \neq j$. Consequently, in this case the result is equal to 4.

Of course, a schoolmate can't date with him/herself.

Help **HamsterW** and write the program that will find the result by the given sequence $t_1, t_2, \ldots, t_n$.

Maybe one of you will be in these matched cases.

Input

The first line of the input data contains an integer n $(1 \leq n \leq 5 * 10^5)$ which represents the number of registered schoolmates of the Five-day lovers.

The second line contains a sequence of integers$t_1, t_2, \ldots, t_n(-20 \leq t_i \leq 20)$, t_i is the parameter of the $i-th$ schoolmate that has been assigned to the schoolmate by the result of the analysis of his interests.

Output

Print the number of matches of schoolmates with opposite t. The opposite number for x is number $-x$ (0 is opposite to itself).

Example

standard input	standard output
6 -1 1 0 0 1 0	5

Problem C. Can You Help ZSGW

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Finally, the number of total active cases in the city where ZSGW lives has fallen to zero!

As a celebration, ZSGW decided to go shopping and fetch a lot of delicious food. To reduce the risk of being infected, he decided to fill his shopping cart as much as possible so that he will need fewer times to go outside.

The shopping cart of the supermarket is very special, which can be regarded as a stack storing goods from bottom to the top. That is, the item in the cart are lined up from bottom to the top, where the first put in the cart lies on the bottom and the last one is on the top.

Moreover, the prices of items on a row of a shelf form a permutation, which means if there are n items on the shelf, the prices will be identical integers from 1 to n.

Standing in front of the shelf containing various goods, ZSGW started to pick up what he wanted one by one. He didn't want to spend too much money, so he selected the goods as the following rule:

1. Suppose all the goods lies in a row on the shelf, initially ZSGW stands at the left of the shelf.
2. He will consider each item one by one from left to the right.
3. When he stands in front of one item, he will compare the price of the item to that of the item at the top of his shopping cart. If the price of the top item is greater than the current item, he will remove it from the cart. He will continue removing the top item of the cart until the price of the top item is less than the price of item currently in front of him or the cart is empty. Then he will put this item into the shopping cart and move to the next item.

For Example, there are items in a row whose prices are:$\{2, 1, 4, 5, 3, 6\}$, the whole procedure for ZSGW to pick up goods will be:

1. Move to the first item. Due to the cart is empty, he will directly put it in. The item in cart will be $\{2\}$.
2. After considering the second one, the first item will be removed. The item in cart will be $\{1\}$.
3. For the third one, because 4 is greater than the top item in cart(which is 1), so the cart will be $\{1, 4\}$.
4. Because $4 < 5$, the cart now will be $\{1, 4, 5\}$.
5. Now it comes to the fifth item, because we have $5 > 3$ and $4 > 3$, these two items will both be removed before put in the fifth item. So the cart will be $\{1, 3\}$.
6. Finally, the items in the cart will be $\{1, 3, 6\}$.

ZSGW also took notes when he make shopping. He marked down the number of items every time after he put one item into the cart. As for the case above, the numbers he marked down will be: $\{1, 1, 2, 3, 2, 3\}$.

He believed that with his notes he can recover the price of each item later, only to find that he even can't recognize some of his own handwriting! As a brilliant programmer, ZSGW hopes that you can help him to figure out the price of each item.

Input

The input will contains multiple cases. The first line starts with an integer T, the number of test cases.

For each test cases, the first line will be an integer n, the number of items on the shelf.

The next line contains n positive integers or -1 separated by spaces, which are the numbers marked by ZSGW. Additionally, -1 means that ZSGW can't tell the exact number of this position.

For all test cases, $1 \leq n \leq 10^5$, and the sum of n no more than $2 * 10^5$.($\sum n \leq 2 * 10^5$)

Output

For each test case, print a line consists of n numbers, which is a solution of the prices.

NOTE that your solution must be a permutation from 1 to n, and if there exists multiple solutions, you need to output the solution which is the lexicographically smallest.

Here permutation p_1 of length n is lexicographically smaller than permutation p_2 if and only if there exists an index i, $1 \leq i \leq n-1$ that $p_1[j] = p_2[j], 1 \leq j \leq i$ and $p_1[i] < p_2[i]$.

It's guaranteed that there are always at least one solution for each test case.

Example

standard input	standard output
4 6 1 1 2 3 2 3 6 1 -1 2 -1 2 -1 6 -1 1 -1 3 -1 3 3 -1 1 1	2 1 4 5 3 6 1 4 3 5 2 6 2 1 3 5 6 4 3 2 1

Problem D. DIY Masks at Home

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

During the outbreak of COVID-19, xyjj has to stay at home for a long time, which is because he doesn't have any mask. If he wants to go outside, he must wear a mask to avoid being infected. But if he wants to have a mask, he must go outside to buy one!

Finally, he came up with a solution: Why not DIY at home! Quickly he found the materials required: scissors, needle and threads, and a large piece of cloth (cut from curtains of his room). To make a mask, he need to cut out a piece of square clothes, and other part of the mask can be easily made so it doesn't matter. The original cloth can be regarded as a rectangle of $n \times m$ cm^2 , which is made up of 1×1 small colorful squares as the image showing below. (That's a typical programmer's style, isn't it?)

However, xyjj doesn't want his mask to be colorful, so he want the square he selected to have only one single color. Meanwhile, he wants the square to be cut out as large as possible, or the mask might not be able to cover his face. Can you help him with this problem?

Input

The input consists of several test cases. The first line contains an integer T, the number of test cases.

For each test case:

The first line will be two integers n, m, which describe the size of the cloth.

For the next n lines, each line will be a string of m upper-case Latin letters, which describe the pattern of the cloth. Each letter refers to a kind of color and places which have the same letters also have the same color.

Here it's guaranteed that $1 \leq n, m \leq 2.000$, and $\sum n, \sum m \leq 2,000$ for all test cases.

Output

For each test case, print one integer ans in a single line, which is the maximum length of the square we can get. (i.e the result square's size will be $ans \times ans$ cm^2).

Example

standard input	standard output
1 3 4 AABB AACC CCCC	2

Problem E. E Yu is a Brutal Creature

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

During the outbreak of the COVID-19, **HamsterW** was bored at home, and he chose to pass the time by reading comics. He read a manga drawn by **E Yu**, and in the process of reading the manga he saw a story like this:

As we all known, there is a creature called **E Yu** who doesn't have heart.

Tanjiro Kamado is a fierce demon leader. He leads demons and the total number of demons is N.

He often leads demons to rob the traders. But **Tanjiro** is a fair leader, after each robbery, he will distribute the treasures robbed **equally to himself and his N demons (i.e. There will be $N+1$ members in total involved in the distribution!)**. The number of treasures as V.

Tanjiro has the ability to transform humans into demons. After each robbery, his demon army will **add a demon**.

Tanjiro is different from the previous leader. He is no longer afraid of the sun. So his robbery efficiency is greatly improved. In the frequent robbery, **Tanjiro** found that the total number of treasures V had a certain relationship with the square of his army size N, formally is: $V = N^2 + 1$.

The demons are very brutal. Once the treasures cannot be divided equally among **Tanjiro** and N demons, there will be conflict between them.

For the harmony of the team, **Tanjiro** has to resolve conflicts. Smartly, he already knows ho to calculate whether there will be a conflict after each robbery.

But **Tanjiro** wants to go further. He wants to know how many conflicts (the number is C) will occur when his demon army grows from 0 to N.

Because **Tanjiro** is a very gentle boy, **HamsterW** want to help him calculate this result.

Input

The input contains multiple cases. The first line contains an nonnegative integer T the number of test cases.

Each case has one line containing a **nonnegative integer** N.

It is guaranteed $0 \le N \le 1\,000\,000, \quad 1 \le T \le 1\,000$.

Output

For each test case, print a number C in one line, denoting the number of conflicts occurring when his demon army grows from 0 to N.

Example

standard input	standard output
2 2 13	1 12