A - Five Integers

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 100 points

Problem Statement

Print how many distinct integers there are in given five integers A, B, C, D, and E.

Constraints

- $0 \leq A, B, C, D, E \leq 100$
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

A B C D E

Output

Print the answer.

Sample Input 1

```
31 9 24 31 24
```

Sample Output 1

```
3
```

In the given five integers $31, 9, 24, 31$, and 24, there are three distinct integers $9, 24$, and 31. Thus, 3 should be printed.

Sample Input 2

```
0 0 0 0 0
```

Sample Output 2

```
1
```

B - Prefix?

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 200 points

Problem Statement

You are given two strings S and T consisting of lowercase English letters. Determine if S is a prefix of T.

▶ What is a prefix?

Constraints

- S and T are strings of lengths between 1 and 100 (inclusive) consisting of lowercase English letters.

Input

Input is given from Standard Input in the following format:

```
S
T
```

Output

Print `Yes` if S is a prefix of T; print `No` otherwise. Note that the judge is case-sensitive.

Sample Input 1

```
atco
atcoder
```

Sample Output 1

```
Yes
```

`atco` is a prefix of `atcoder`. Thus, `Yes` should be printed.

Sample Input 2

```
code
atcoder
```

Sample Output 2

```
No
```

`code` is not a prefix of `atcoder`. Thus, `No` should be printed.

Sample Input 3

```
abc
abc
```

Sample Output 3

```
Yes
```

Note that a string is also a prefix of itself.

Sample Input 4

```
aaaa
aa
```

Sample Output 4

```
No
```

C - Chinese Restaurant

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 300 points

Problem Statement

Person 0, Person 1, . . ., and Person $(N-1)$ are sitting around a turntable in their counterclockwise order, evenly spaced. Dish p_i is in front of Person i on the table.
You may perform the following operation 0 or more times:

- Rotate the turntable by one N-th of a counterclockwise turn. As a result, the dish that was in front of Person i right before the rotation is now in front of Person $(i+1) \bmod N$.

When you are finished, Person i is happy if Dish i is in front of Person $(i-1) \bmod N$, Person i, or Person $(i+1) \bmod N$.
Find the maximum possible number of happy people.

▶ What is $a \bmod m$?

Constraints

- $3 \leq N \leq 2 \times 10^5$
- $0 \leq p_i \leq N-1$
- $p_i \neq p_j$ if $i \neq j$.
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

N
$p_0 \ \cdots \ p_{N-1}$

Output

Print the answer.

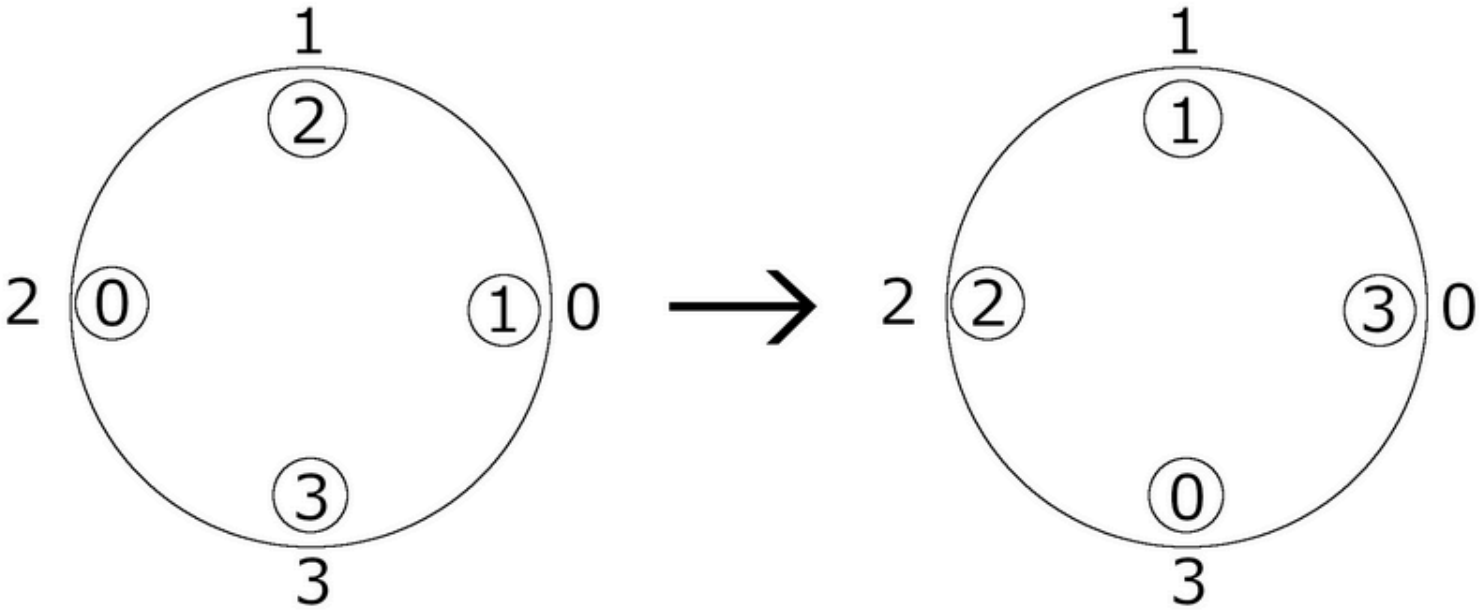
Sample Input 1

```
4
1 2 0 3
```

Sample Output 1

```
4
```

The figure below shows the table after one operation.



Here, there are four happy people:

- Person 0 is happy because Dish 0 is in front of Person 3 $(= (0 - 1) \bmod 4)$;
- Person 1 is happy because Dish 1 is in front of Person 1 $(= 1)$;
- Person 2 is happy because Dish 2 is in front of Person 2 $(= 2)$;
- Person 3 is happy because Dish 3 is in front of Person 0 $(= (3 + 1) \bmod 4)$.

There cannot be five or more happy people, so the answer is 4.

Sample Input 2

```
3
0 1 2
```

Sample Output 2

```
3
```

Sample Input 3

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

Sample Output 3

```
5
```

D - Unique Username

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 400 points

Problem Statement

Takahashi is having trouble with deciding a username for a service. Write a code to help him.

Find a string X that satisfies all of the following conditions:

- X is obtained by the following procedure:
  - Let $S'_1, S'_2, \ldots, S'_N$ be a permutation of $S_1, S_2, \ldots, S_N$. Let X be the concatenation of S'_1, (1 or more copies of _), S'_2, (1 or more copies of _), . . ., (1 or more copies of _), and S'_N, in this order.
- The length of X is between 3 and 16, inclusive.
- X does not coincide with any of M strings $T_1, T_2, \ldots, T_M$.

If there is no X that satisfies all of the conditions, print `-1` instead.

Constraints

- $1 \leq N \leq 8$
- $0 \leq M \leq 10^5$
- N and M are integers.
- $1 \leq |S_i| \leq 16$
- $N - 1 + \sum |S_i| \leq 16$
- $S_i \neq S_j$ if $i \neq j$.
- S_i is a string consisting of lowercase English letters.
- $3 \leq |T_i| \leq 16$
- $T_i \neq T_j$ if $i \neq j$.
- T_i is a string consisting of lowercase English letters and _.

Input

Input is given from Standard Input in the following format:

$$
\begin{aligned}
&N \quad M\\
&S_1\\
&S_2\\
&\vdots\\
&S_N\\
&T_1\\
&T_2\\
&\vdots\\
&T_M
\end{aligned}
$$

Output

Print a string X that satisfies all of the conditions. If there is no X that satisfies all of the conditions, print `-1` instead.

If there are multiple solutions, print any of them.

Sample Input 1

```
1 1
chokudai
chokudai
```

Sample Output 1

```
-1
```

The only string that satisfies the first and second conditions is $X =$ `chokudai`, but it coincides with T_1.

Thus, there is no X that satisfies all of the conditions, so `-1` should be printed.

Sample Input 2

```
2 2
choku
dai
chokudai
choku_dai
```

Sample Output 2

```
dai_choku
```

Strings like `choku__dai` (which has two _'s between `choku` and `dai`) also satisfy all of the conditions.

Sample Input 3

```
2 2
chokudai
atcoder
chokudai_atcoder
atcoder_chokudai
```

Sample Output 3

```
-1
```

`chokudai__atcoder` and `atcoder__chokudai` (which have two _'s between `chokudai` and `atcoder`) have a length of 17, which violates the second condition.

Sample Input 4

```
4 4
ab
cd
ef
gh
hoge
fuga
____
_ab_cd_ef_gh_
```

Sample Output 4

```
ab__ef___cd_gh
```

The given T_i may contain a string that cannot be obtained by the procedure described in the first condition.

E - Chinese Restaurant (Three-Star Version)

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 500 points

Problem Statement

Person 0, Person 1, . . ., and Person $(N-1)$ are sitting around a turntable in counterclockwise order, evenly spaced. Dish p_i is in front of Person i on the table.

You may perform the following operation 0 or more times:

- Rotate the turntable by one N-th of a counterclockwise turn. The dish that was in front of Person i right before the rotation is now in front of Person $(i+1) \bmod N$.

When you are finished, Person i gains frustration of k, where k is the minimum integer such that Dish i is in front of either Person $(i-k) \bmod N$ or Person $(i+k) \bmod N$.

Find the minimum possible sum of frustration of the N people.

▶ What is $a \bmod m$?

Constraints

- $3 \leq N \leq 2 \times 10^5$
- $0 \leq p_i \leq N-1$
- $p_i \neq p_j$ if $i \neq j$.
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

N
$p_0 \quad \cdots \quad p_{N-1}$

Output

Print the answer.

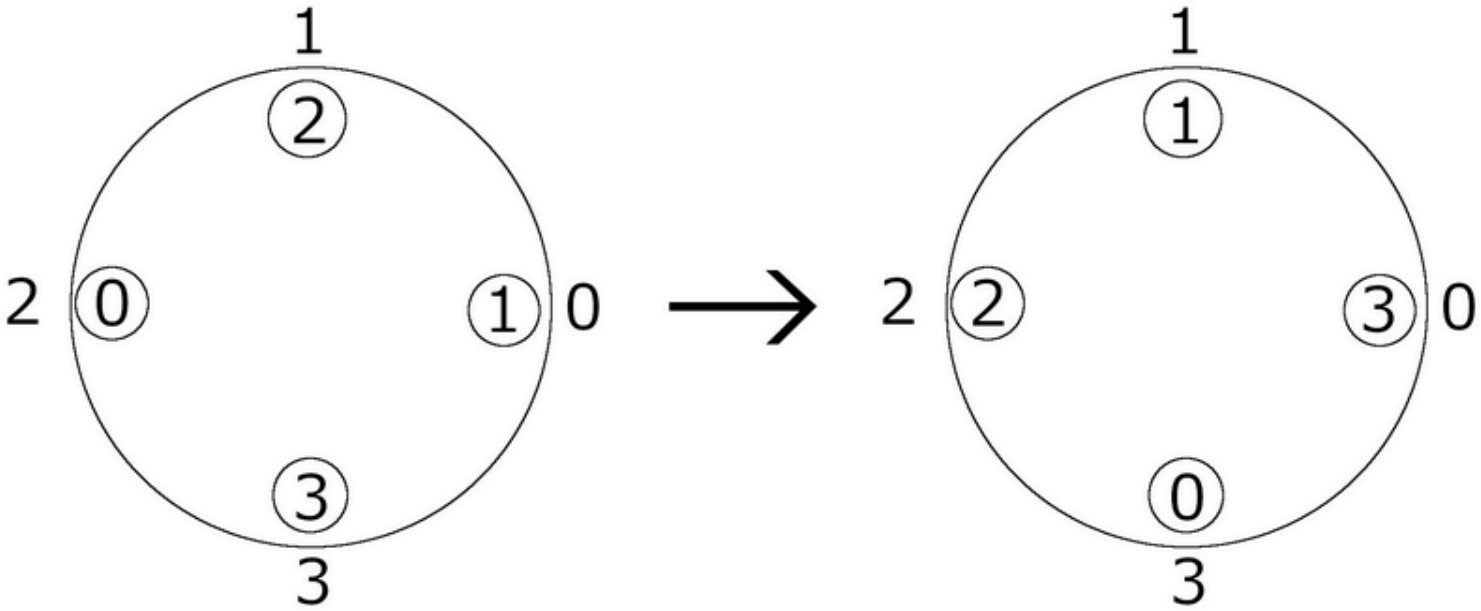
Sample Input 1

```
4
1 2 0 3
```

Sample Output 1

```
2
```

The figure below shows the table after one operation.



Here, the sum of their frustration is 2 because:

- Person 0 gains a frustration of 1 since Dish 0 is in front of Person 3 $(= (0 - 1) \bmod 4)$;
- Person 1 gains a frustration of 0 since Dish 1 is in front of Person 1 $(= (1 + 0) \bmod 4)$;
- Person 2 gains a frustration of 0 since Dish 2 is in front of Person 2 $(= (2 + 0) \bmod 4)$;
- Person 3 gains a frustration of 1 since Dish 3 is in front of Person 0 $(= (3 + 1) \bmod 4)$.

We cannot make the sum of their frustration less than 2, so the answer is 2.

Sample Input 2

```
3
0 1 2
```

Sample Output 2

```
0
```

Sample Input 3

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

Sample Output 3

```
20
```

F - Best Concatenation

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 500 points

Problem Statement

You are given N strings $S_1, S_2, \ldots, S_N$ consisting of digits from 1 through 9 and the character x.

We will choose a permutation $P = (P_1, P_2, \ldots, P_N)$ of $(1, 2, \ldots, N)$ to construct a string $T = S_{P_1} + S_{P_2} + \cdots + S_{P_N}$, where $+$ denotes a concatenation of strings.

Then, we will calculate the "score" of the string $T = T_1 T_2 \ldots T_{|T|}$ (where $|T|$ denotes the length of T). The score is calculated by the following 9 steps, starting from the initial score 0:

- Add 1 point to the score as many times as the number of integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ x, and $T_j =$ 1.
- Add 2 points to the score as many times as the number of integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ x, and $T_j =$ 2.
- Add 3 points to the score as many times as the number of integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ x, and $T_j =$ 3.
- $\cdots$
- Add 9 points to the score as many times as the number of integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ x, and $T_j =$ 9.

Find the maximum possible score of T when P can be chosen arbitrarily.

Constraints

- $2 \leq N \leq 2 \times 10^5$
- N is an integer.
- S_i is a string of length at least 1 consisting of digits from 1 through 9 and the character x.
- The sum of lengths of $S_1, S_2, \ldots, S_N$ is at most 2×10^5.

Input

Input is given from Standard Input in the following format:

$$
\begin{aligned}
&N\\
&S_1\\
&S_2\\
&\vdots\\
&S_N
\end{aligned}
$$

Output

Print the answer.

Sample Input 1

```
3
1X3
59
XXX
```

Sample Output 1

```
71
```

When $P = (3, 1, 2)$, we have $T = S_3 + S_1 + S_2 =$ XXX1X359. Then, the score of T is calculated as follows:

- there are 3 integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ X, and $T_j =$ 1;
- there are 4 integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ X, and $T_j =$ 3;
- there are 4 integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ X, and $T_j =$ 5;
- there are 4 integer pairs (i, j) such that $1 \leq i < j \leq |T|, T_i =$ X, and $T_j =$ 9.

Therefore, the score of T is $1 \times 3 + 3 \times 4 + 5 \times 4 + 9 \times 4 = 71$, which is the maximum possible value.

Sample Input 2

```
10
X63X395XX
X2XX3X22X
13
3716XXX6
45X
X6XX
9238
281X92
1XX4X4XX6
54X9X711X1
```

Sample Output 2

```
3010
```

G - Random Student ID

Time Limit: 3 sec / Memory Limit: 1024 MB

Score : 600 points

Problem Statement

Takahashi Elementary School has N new students. For $i = 1, 2, \ldots, N$, the name of the i-th new student is S_i (which is a string consisting of lowercase English letters). The names of the N new students are distinct.

The N students will be assigned a student ID $1, 2, 3, \ldots, N$ in **ascending lexicographical order** of their names. However, instead of the ordinary order of lowercase English letters where a is the minimum and z is the maximum, we use the following order:

- First, Principal Takahashi chooses a string P from the $26!$ permutations of the string `abcdefghijklmnopqrstuvwxyz` of length 26, uniformly at random.
- The lowercase English characters that occur earlier in P are considered smaller.

For each of the N students, find the expected value, modulo 998244353, of the student ID assigned (see Notes).

▶ What is the lexicographical order?

Notes

We can prove that the sought expected value is always a rational number. Moreover, under the Constraints of this problem, when the value is represented as $\frac{P}{Q}$ by two coprime integers P and Q, we can prove that there is a unique integer R such that $R \times Q \equiv P \pmod{998244353}$ and $0 \leq R < 998244353$. Find such R.

Constraints

- $2 \leq N$
- N is an integer.
- S_i is a string of length at least 1 consisting of lowercase English letters.
- The sum of lengths of the given strings is at most 5×10^5.
- $i \neq j \Rightarrow S_i \neq S_j$

Input

Input is given from Standard Input in the following format:

$$
\begin{aligned}
&N\\
&S_1\\
&S_2\\
&\vdots\\
&S_N
\end{aligned}
$$

Output

Print N lines. For each $i = 1, 2, \ldots, N$, the i-th line should contain the expected value, modulo 998244353, of the student ID assigned to Student i.

Sample Input 1

```
3
a
aa
ab
```

Sample Output 1

```
1
499122179
499122179
```

The expected value of the student ID assigned to Student 1 is 1; the expected values of the student ID assigned to Student 2 and 3 are $\frac{5}{2}$.

Note that the answer should be printed modulo 998244353. For example, the sought expected value for Student 2 and 3 is $\frac{5}{2}$, and we have $2 \times 499122179 \equiv 5 \pmod{998244353}$, so 499122179 should be printed.

Sample Input 2

```
3
a
aa
aaa
```

Sample Output 2

```
1
2
3
```

Ex - Taboo

Time Limit: 4 sec / Memory Limit: 1024 MB

Score : 600 points

Problem Statement

You are given a string S. Takahashi may perform the following operation 0 or more times:

- Choose an integer i such that $1 \leq i \leq |S|$ and change the i-th character of S to $*$.

Takahashi's objective is to make S not contain any of N strings $T_1, T_2, \ldots, T_N$ **as a substring**.
Find the minimum number of operations required to achieve the objective.

Constraints

- $1 \leq |S| \leq 5 \times 10^5$
- $1 \leq N$
- N is an integer.
- $1 \leq |T_i|$
- $\sum |T_i| \leq 5 \times 10^5$
- $T_i \neq T_j$ if $i \neq j$.
- S and T_i are strings consisting of lowercase English letters.

Input

Input is given from Standard Input in the following format:

$$
\begin{aligned}
&S\\
&N\\
&T_1\\
&T_2\\
&\vdots\\
&T_N
\end{aligned}
$$

Output

Print the answer.

Sample Input 1

```
abcdefghijklmn
3
abcd
ijk
ghi
```

Sample Output 1

```
2
```

If he performs the operation twice by choosing 1 and 9 for i, S becomes `*bcdefgh*jklmn`; now it does not contain `abcd`, `ijk`, or `ghi` as a substring.

Sample Input 2

```
atcoderbeginnercontest
1
abc
```

Sample Output 2

```
0
```

No operation is needed.

Sample Input 3

```
aaaaaaaaa
2
aa
xyz
```

Sample Output 3

```
4
```