

# Longest Common Subsequence

Time Limit: 1

You are given  $n$  strings, each a permutation of the first  $k$  upper-case letters of the alphabet.

String  $s$  is a *subsequence* of string  $t$  if and only if it is possible to delete some (possibly zero) characters from the string  $t$  to get the string  $s$ .

Compute the length of the longest common *subsequence* of all  $n$  strings.

## Input

The first line of input contains two integers  $n$  ( $1 \leq n \leq 10^5$ ) and  $k$  ( $1 \leq k \leq 26$ ), where  $n$  is the number of strings, and the strings are all permutations of the first  $k$  upper-case letters of the alphabet.

Each of the next  $n$  lines contains a single string  $t$ . It is guaranteed that every  $t$  contains each of the first  $k$  upper-case letters of the alphabet exactly once.

## Output

Output a single integer, the length of the longest subsequence that appears in all  $n$  strings.

### Sample Input 1

```
2 3
BAC
ABC
```

### Sample Output 1

```
2
```

### Sample Input 2

```
3 8
HGBDFCAE
ADBGHFCE
HCFGBD AE
```

### Sample Output 2

```
3
```

### Sample Input 3

```
6 8
AHFBGDCE
FABGCEHD
AHDGFBCE
DABHGCFE
ABCHFEDG
DGABHFCE
```

### Sample Output 3

```
4
```