[A - Scarborough Fair](https://cn.vjudge.net/problem/CodeForces-897A" \t "_blank)

 Are you going to Scarborough Fair?

Parsley, sage, rosemary and thyme.

Remember me to one who lives there.

He once was the true love of mine.

Willem is taking the girl to the highest building in island No.28, however, neither of them knows how to get there.

Willem asks his friend, Grick for directions, Grick helped them, and gave them a task.

Although the girl wants to help, Willem insists on doing it by himself.

Grick gave Willem a string of length n.

Willem needs to do m operations, each operation has four parameters l, r, c1, c2, which means that all symbols c1 in range [l, r] (from l-th to r-th, including l and r) are changed into c2. String is 1-indexed.

Grick wants to know the final string after all the m operations.

**Input**

The first line contains two integers n and m (1 ≤ n, m ≤ 100).

The second line contains a string s of length n, consisting of lowercase English letters.

Each of the next m lines contains four parameters l, r, c1, c2 (1 ≤ l ≤ r ≤ n, c1, c2 are lowercase English letters), separated by space。

**Output**

Output string s after performing m operations described above.

**Example**

**Input**

3 1  
ioi  
1 1 i n

**Output**

noi

**Input**

5 3  
wxhak  
3 3 h x  
1 5 x a  
1 3 w g

**Output**

gaaak

**Note**

For the second example:

After the first operation, the string is wxxak.

After the second operation, the string is waaak.

After the third operation, the string is gaaak.

[B - Wc与整数分组](https://cn.vjudge.net/problem/NBUT-1722)

将前n个正整数分成k组，使得每组中的正整数两两互质。求k的最小值。

注意：两个正整数互质，指的是两个正整数的最大公因数等于1。

**Input**

第一行有一个整数T (1<=T<=1e5)，表示数据组数，对于每组数据：   
每一行只有一个整数n(1<=n<=1e9)

**Output**

每组数据输出一行一个整数，表示k的最小值

**Sample Input**

2

3

4

**Sample Output**

1

2

**Hint**

对于第一组数据，由于1，2，3这三个数本来就两两互质，所以只要分成一组即可，即k=1。

对于第二组数据，我们可以将1，2，3，4这四个数分成1，4和2，3两组，即k=2,就能满足要求。容易发现这也是我们能获得的最小的k。

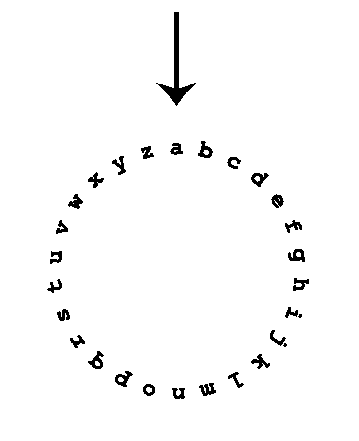
如果cin超时可以使用scanf

如果scanf使用long long ，必须使用%I64d

[C - Night at the Museum](https://cn.vjudge.net/problem/CodeForces-731A)

Grigoriy, like the hero of one famous comedy film, found a job as a night security guard at the museum. At first night he received embosser and was to take stock of the whole exposition.

Embosser is a special devise that allows to "print" the text of a plastic tape. Text is printed sequentially, character by character. The device consists of a wheel with a lowercase English letters written in a circle, static pointer to the current letter and a button that print the chosen letter. At one move it's allowed to rotate the alphabetic wheel one step clockwise or counterclockwise. Initially, static pointer points to letter 'a'. Other letters are located as shown on the picture:



After Grigoriy add new item to the base he has to print its name on the plastic tape and attach it to the corresponding exhibit. It's not required to return the wheel to its initial position with pointer on the letter 'a'.

Our hero is afraid that some exhibits may become alive and start to attack him, so he wants to print the names as fast as possible. Help him, for the given string find the minimum number of rotations of the wheel required to print it.

**Input**

The only line of input contains the name of some exhibit — the non-empty string consisting of no more than 100 characters. It's guaranteed that the string consists of only lowercase English letters.

**Output**

Print one integer — the minimum number of rotations of the wheel, required to print the name given in the input.

**Example**

**Input**

zeus

**Output**

18

**Input**

map

**Output**

35

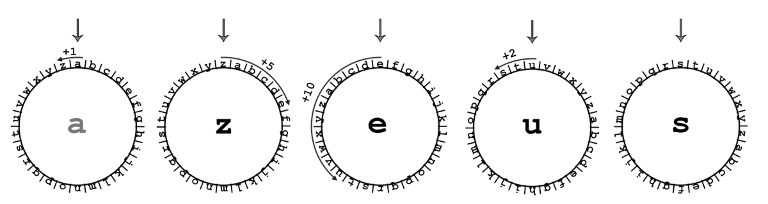
**Input**

ares

**Output**

34

**Note**



To print the string from the first sample it would be optimal to perform the following sequence of rotations:

1. from 'a' to 'z' (1 rotation counterclockwise),
2. from 'z' to 'e' (5 clockwise rotations),
3. from 'e' to 'u' (10 rotations counterclockwise),
4. from 'u' to 's' (2 counterclockwise rotations).

In total, 1 + 5 + 10 + 2 = 18 rotations are required.

[D - 最小公倍数挑战](https://cn.vjudge.net/problem/51Nod-1419)

几天以前，我学习了最小公倍数。玩得挺久了，想换换口味。

我不想用太多的数字，我想从1到n中选三个数字（可以相同）。使得他们的最小公倍数最大。

**Input**

单组测试数据。   
第一行有一个整数n (1≤n≤1,000,000)。

**Output**

输出一个整数表示选三个数字的最大的最小公倍数。

**Sample Input**

9

7

**Sample Output**

504

210

[E - IP查询](https://cn.vjudge.net/problem/NBUT-1682)

我们知道现实生活中每一个座城市对应一个IP段。现在为了简化问题，将IP端直接看作一个整型数，每座城市也有自己唯一的标识ID，也可以看做一个整数。那么问题来了，现在已知有多个闭区间代表多个IP段，每个区间对应一个城市的ID。现在Gealo要查询某个IP属于哪个城市，希望你们来帮他完成。

**Input**

第一行输入t，表示有t组数据(t <= 5)   
接下来一行输入n，表示有n个区间(0 <= n<= 10^5)   
接下来n行，每行输入三个整数x,y,id。代表区间[x,y]对应的城市ID。数据确保两个区间的交集为空，且ID唯一。(0 <=x<y<=10^8 , 0<= ID <= 10^8)   
接下来一行输入一个整数m，代表m次查询(0 <= m <= 10^5)   
接下来m行，每行输入一个整数x，代表所查询的IP(x <= 10^8)

**Output**

对于每次查询，输出一行，表示其对应的城市ID。   
如果未找到就输出-1。

**Sample Input**

1

2

3 5 99

1 2 77

3

1

3

9

**Sample Output**

77

99

-1

**Hint**

如果cin超时可以使用scanf

如果scanf使用long long ，必须使用%I64d

[F - Wc的朗利特树篱迷宫(Ⅲ)](https://cn.vjudge.net/problem/NBUT-1721)

Wc的暑假一直持续到了寒假，这次他和他的麻瓜朋友们一起去英国的威尔特郡，参观朗利特树篱迷宫。WC为了增加迷宫的乐趣，特意给迷宫施加了一个随机魔法。

每次进入迷宫时，迷宫都会重新规划，但可以肯定的是迷宫一定只有一个起点和一个终点。

这次Wc为了彰显自己的能力，还在迷宫中加了其他奇怪的东西。

迷宫的围墙中可能会有且只有一扇门，需要k把钥匙才能开启，钥匙分布在迷宫的各个地方，且数量不定。

现在给你几个迷宫，让你判断一下WC和他的麻瓜朋友们能否走出迷宫。由于朋友比较麻瓜，他们只会朝上下左右四个方向走。

‘S’ 表示起点

‘E’ 表示终点

‘\*’ 表示路，可以走

‘#’ 表示墙，不能行走

‘M’ 表示门，如果集齐k把要是即可成为‘\*’可以走

‘K’ 表示钥匙

**Input**

第一行为一个正整数T，表示接下来又T个迷宫   
对于每个迷宫，第一行为两个个正整数n，表示迷宫的长和宽，(1<=n<=100)，k，表示需要k把钥匙才能打开门   
接下来会有n行，每行仅由‘S’、‘E’、’\*’、’#’、‘K‘、’M‘组成，表示迷宫。

**Output**

对于每个迷宫，如果可以从起点到终点，则输出‘YES’，否则输出‘NO’。

**Sample Input**

3

5 1

S\*\*K#

\*\*\*#\*

\*\*M\*\*

\*#\*\*\*

#\*\*\*E

5 1

S\*\*\*#

\*\*\*#K

\*\*M\*\*

\*#\*\*\*

#\*\*\*E

5 3

S\*\*K#

\*\*\*#\*

\*\*M\*\*

\*#\*\*\*

\*\*\*\*E

**Sample Output**

YES

NO

YES

[G - Tic-Tac-Toe](https://cn.vjudge.net/problem/FZU-2283)

Kim likes to play Tic-Tac-Toe.

Given a current state, and now Kim is going to take his next move. Please tell Kim if he can win the game in next 2 moves if both player are clever enough.

Here “next 2 moves” means Kim’s 2 move. (Kim move,opponent move, Kim move, stop).



Game rules:

Tic-tac-toe (also known as noughts and crosses or Xs and Os) is a paper-and-pencil game for two players, X and O, who take turns marking the spaces in a 3×3 grid. The player who succeeds in placing three of their marks in a horizontal, vertical, or diagonal row wins the game.

**Input**

First line contains an integer T (1 ≤ T ≤ 10), represents there are T test cases.

For each test case: Each test case contains three lines, each line three string(“o” or “x” or “.”)(All lower case letters.)

x means here is a x

o means here is a o

. means here is a blank place.

Next line a string (“o” or “x”) means Kim is (“o” or “x”) and he is going to take his next move.

**Output**

For each test case:

If Kim can win in 2 steps, output “Kim win!”

Otherwise output “Cannot win!”

**Sample Input**

3

. . .

. . .

. . .

o

o x o

o . x

x x o

x

o x .

. o .

. . x

o

**Sample Output**

Cannot win!

Kim win!

Kim win!