Codeforces Round #373 (Div. 2)

A. Vitya in the Countryside

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Every summer Vitya comes to visit his grandmother in the countryside. This summer, he got a huge wart. Every grandma knows that one should treat warts when the moon goes down. Thus, Vitya has to catch the moment when the moon is down.

Moon cycle lasts 30 days. The size of the visible part of the moon (in Vitya's units) for each day is $0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1$, and then cycle repeats, thus after the second 1 again goes 0.

As there is no internet in the countryside, Vitya has been watching the moon for n consecutive days and for each of these days he wrote down the size of the visible part of the moon. Help him find out whether the moon will be up or down next day, or this cannot be determined by the data he has.

Input

The first line of the input contains a single integer n $(1 \le n \le 92)$ — the number of consecutive days Vitya was watching the size of the visible part of the moon.

The second line contains n integers a_i $(0 \le a_i \le 15)$ — Vitya's records.

It's guaranteed that the input data is consistent.

Output

If Vitya can be sure that the size of visible part of the moon on day $n+1$ will be less than the size of the visible part on day n, then print "`DOWN`" at the only line of the output. If he might be sure that the size of the visible part will increase, then print "`UP`". If it's impossible to determine what exactly will happen with the moon, print `-1`.

Examples

input

```
5
3 4 5 6 7
```

output

```
UP
```

input

```
7
12 13 14 15 14 13 12
```

output

```
DOWN
```

input

```
1
8
```

output

```
-1
```

Note

In the first sample, the size of the moon on the next day will be equal to 8, thus the answer is "`UP`".

In the second sample, the size of the moon on the next day will be 11, thus the answer is "`DOWN`".

In the third sample, there is no way to determine whether the size of the moon on the next day will be 7 or 9, thus the answer is `-1`.

B. Anatoly and Cockroaches

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Anatoly lives in the university dorm as many other students do. As you know, cockroaches are also living there together with students. Cockroaches might be of two colors: black and red. There are n cockroaches living in Anatoly's room.

Anatoly just made all his cockroaches to form a single line. As he is a perfectionist, he would like the colors of cockroaches in the line to **alternate**. He has a can of black paint and a can of red paint. In one turn he can either swap any two cockroaches, or take any single cockroach and change it's color.

Help Anatoly find out the minimum number of turns he needs to make the colors of cockroaches in the line alternate.

Input

The first line of the input contains a single integer n $(1 \le n \le 100\,000)$ — the number of cockroaches.

The second line contains a string of length n, consisting of characters '`b`' and '`r`' that denote black cockroach and red cockroach respectively.

Output

Print one integer — the minimum number of moves Anatoly has to perform in order to make the colors of cockroaches in the line to alternate.

Examples

input

```
5
rbbrr
```

output

```
1
```

input

```
5
bbbbb
```

output

```
2
```

input

```
3
rbr
```

output

```
0
```

Note

In the first sample, Anatoly has to swap third and fourth cockroaches. He needs 1 turn to do this.

In the second sample, the optimum answer is to paint the second and the fourth cockroaches red. This requires 2 turns.

In the third sample, the colors of cockroaches in the line are alternating already, thus the answer is 0.

C. Efim and Strange Grade

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Efim just received his grade for the last test. He studies in a special school and his grade can be equal to any positive decimal fraction. First he got disappointed, as he expected a way more pleasant result. Then, he developed a tricky plan. Each second, he can ask his teacher to round the grade at any place after the decimal point (also, he can ask to round to the nearest integer).

There are t seconds left till the end of the break, so Efim has to act fast. Help him find what is the maximum grade he can get in no more than t seconds. Note, that he can choose to not use all t seconds. Moreover, he can even choose to not round the grade at all.

In this problem, classic rounding rules are used: while rounding number to the n-th digit one has to take a look at the digit $n+1$. If it is less than 5 than the n-th digit remain unchanged while all subsequent digits are replaced with 0. Otherwise, if the $n+1$ digit is greater or equal to 5, the digit at the position n is increased by 1 (this might also change some other digits, if this one was equal to 9) and all subsequent digits are replaced with 0. At the end, all trailing zeroes are thrown away.

For example, if the number 1.14 is rounded to the first decimal place, the result is 1.1, while if we round 1.5 to the nearest integer, the result is 2. Rounding number 1.299996121 in the fifth decimal place will result in number 1.3.

Input

The first line of the input contains two integers n and t ($1 \le n \le 200\ 000$, $1 \le t \le 10^9$) — the length of Efim's grade and the number of seconds till the end of the break respectively.

The second line contains the grade itself. It's guaranteed that the grade is a positive number, containing at least one digit after the decimal points, and it's representation doesn't finish with 0.

Output

Print the maximum grade that Efim can get in t seconds. Do not print trailing zeroes.

Examples

input

```
6 1
10.245
```

output

```
10.25
```

input

```
6 2
10.245
```

output

```
10.3
```

input

```
3 100
9.2
```

output

```
9.2
```

Note

In the first two samples Efim initially has grade 10.245.

During the first second Efim can obtain grade 10.25, and then 10.3 during the next second. Note, that the answer 10.30 will be considered incorrect.

In the third sample the optimal strategy is to not perform any rounding at all.

E. Sasha and Array

time limit per test: 5 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Sasha has an array of integers $a_1, a_2, ..., a_n$. You have to perform m queries. There might be queries of two types:

1. `1 l r x` — increase all integers on the segment from l to r by values x;
2. `2 l r` — find , where $f(x)$ is the x-th Fibonacci number. As this number may be large, you only have to find it modulo $10^9 + 7$.

In this problem we define Fibonacci numbers as follows: $f(1) = 1, f(2) = 1, f(x) = f(x - 1) + f(x - 2)$ for all $x > 2$.

Sasha is a very talented boy and he managed to perform all queries in five seconds. Will you be able to write the program that performs as well as Sasha?

Input

The first line of the input contains two integers n and m ($1 \le n \le 100\,000$, $1 \le m \le 100\,000$) — the number of elements in the array and the number of queries respectively.

The next line contains n integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le 10^9$).

Then follow m lines with queries descriptions. Each of them contains integers tp_i, l_i, r_i and may be x_i ($1 \le tp_i \le 2$, $1 \le l_i \le r_i \le n$, $1 \le x_i \le 10^9$). Here $tp_i = 1$ corresponds to the queries of the first type and tp_i corresponds to the queries of the second type.

It's guaranteed that the input will contains at least one query of the second type.

Output

For each query of the second type print the answer modulo $10^9 + 7$.

Examples

input

```
5 4
1 1 2 1 1
2 1 5
1 2 4 2
2 2 4
2 1 5
```

output

```
5
7
9
```

Note

Initially, array a is equal to $1, 1, 2, 1, 1$.

The answer for the first query of the second type is $f(1) + f(1) + f(2) + f(1) + f(1) = 1 + 1 + 1 + 1 + 1 = 5$.

After the query `1 2 4 2` array a is equal to $1, 3, 4, 3, 1$.

The answer for the second query of the second type is $f(3) + f(4) + f(3) = 2 + 3 + 2 = 7$.

The answer for the third query of the second type is $f(1) + f(3) + f(4) + f(3) + f(1) = 1 + 2 + 3 + 2 + 1 = 9$.