



🎁 Codeforces celebrates 10 years! We are pleased to announce the crowdfunding-campaign. Congratulate us by the link <https://codeforces.com/10years>.



E. Prefix Enlightenment

time limit per test: 3 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

There are n lamps on a line, numbered from 1 to n . Each one has an initial state off (0) or on (1).

You're given k subsets A_1, \dots, A_k of $\{1, 2, \dots, n\}$, such that the intersection of any three subsets is empty. In other words, for all $1 \leq i_1 < i_2 < i_3 \leq k$, $A_{i_1} \cap A_{i_2} \cap A_{i_3} = \emptyset$.

In one operation, you can choose one of these k subsets and switch the state of all lamps in it. It is guaranteed that, with the given subsets, it's possible to make all lamps be simultaneously on using this type of operation.

Let m_i be the minimum number of operations you have to do in order to make the i first lamps be simultaneously on. Note that there is no condition upon the state of other lamps (between $i + 1$ and n), they can be either off or on.

You have to compute m_i for all $1 \leq i \leq n$.

Input

The first line contains two integers n and k ($1 \leq n, k \leq 3 \cdot 10^5$).

The second line contains a binary string of length n , representing the initial state of each lamp (the lamp i is off if $s_i = 0$, on if $s_i = 1$).

The description of each one of the k subsets follows, in the following format:

Codeforces Round #616 (Div. 2)

Finished

→ Practice?

Want to solve the contest problems after the official contest ends? Just register for practice and you will be able to submit solutions.

Register for practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

The first line of the description contains a single integer c ($1 \leq c \leq n$) — the number of elements in the subset.

The second line of the description contains c distinct integers x_1, \dots, x_c ($1 \leq x_i \leq n$) — the elements of the subset.

It is guaranteed that:

- The intersection of any three subsets is empty;
- It's possible to make all lamps be simultaneously on using some operations.

Output

You must output n lines. The i -th line should contain a single integer m_i — the minimum number of operations required to make the lamps 1 to i be simultaneously on.

Examples

input

Copy

```
7 3
0011100
3
1 4 6
3
3 4 7
2
2 3
```

output

Copy

```
1
2
3
3
3
3
3
3
```

input

Copy

```
8 6
00110011
3
1 3 8
5
1 2 5 6 7
2
6 8
2
3 5
```

→ Problem tags

dfs and similar

dsu

graphs

*2400

No tag edit access

→ Contest materials

• Announcement (en)

• Tutorial (en)

2
4 7
1
2

output

Copy

1
1
1
1
1
1
1
4
4

input

Copy

5 3
00011
3
1 2 3
1
4
3
3 4 5

output

Copy

1
1
1
1
1

input

Copy

19 5
1001001001100000110
2
2 3
2
5 6
2
8 9
5
12 13 14 15 16
1
19

output

Copy

```
0
1
1
1
2
2
2
3
3
3
3
4
4
4
4
4
4
4
5
```

Note

In the first example:

- For $i = 1$, we can just apply one operation on A_1 , the final states will be 1010110;
- For $i = 2$, we can apply operations on A_1 and A_3 , the final states will be 1100110;
- For $i \geq 3$, we can apply operations on A_1 , A_2 and A_3 , the final states will be 1111111.

In the second example:

- For $i \leq 6$, we can just apply one operation on A_2 , the final states will be 11111101;
- For $i \geq 7$, we can apply operations on A_1 , A_3 , A_4 , A_6 , the final states will be 11111111.



ITMO UNIVERSITY