

## Problem Statement

Jack and Daniel are friends.

They want to encrypt their conversation so that they can save themselves from interception by a detective agency. So they invent a new cipher.

Every message is encoded to its binary representation  $B$  of length  $N$ .

Then it is written down  $K$  times, shifted by  $0, 1, \dots, K - 1$  bits.

If  $B = 1001010$  and  $K = 4$  it looks so:

```
1001010
1001010
1001010
1001010
```

Then calculate **XOR** in every column and write it down. This number is called  $S$ . For example, XOR-ing the numbers in the above example results in

```
1110100110
```

Then the encoded message  $S$  and  $K$  are sent to Daniel.

Jack is using this encoding algorithm and asks Daniel to implement a decoding algorithm.

Can you help Daniel implement this?

## Input Format

The first line contains two integers  $N$  and  $K$ .

The second line contains string  $S$  of length  $N + K - 1$  consisting of ones and zeros.

## Output Format

Decoded message of length  $N$ , consisting of ones and zeros.

## Constraints

$$1 \leq N \leq 10^6$$

$$1 \leq K \leq 10^6$$

$$|S| = N + K - 1$$

It is guaranteed that  $S$  is correct.

## Sample Input#00

```
7 4
1110100110
```

## Sample Output#00

```
1001010
```

## Sample Input#01

```
6 2
1110001
```

Sample Output#01

```
101111
```

Explanation

Input#00

```
1001010
1001010
 1001010
   1001010
-----
1110100110
```

Input#01

```
101111
101111
-----
1110001
```