

## Problem I. Partition of Queries

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Richard Roe has just implemented a brand new data structure  $S$ . It is able to perform only two operations: “add” and “query”.

- “add” adds one item to  $S$ . You can assume that this operation takes zero seconds.
- “query” makes some request to  $S$ . This operation takes  $x$  seconds, where  $x$  equals to the number of previously added items.

Other details related to these operations are not important for this problem.

Suddenly Richard understood that he can optimize this structure by rebuilding it from time to time. So he implemented a new function named “rebuild”. This new function works as follows: when “rebuild” is called,  $S$  “forgets” about items added before rebuilding. More precisely, after adding this operation:

- “add” adds one item to  $S$  and takes zero seconds.
- “query” makes some request to  $S$  and takes  $x$  seconds, where  $x$  equals to the number of items added after the last call of “rebuild” (here, assume that a “rebuild” was also called before all queries).
- “rebuild” takes  $y$  seconds.

You are given a sequence of “add” and “query” operations with  $S$ .

Your task is to insert “rebuild” operations in some positions in such a way that the number of seconds all operations will take is minimized.

### Input

The first line of input contains two space-separated integers  $n$  and  $y$  ( $1 \leq n \leq 10^6$ ,  $0 \leq y \leq 10^6$ ).

The second line contains one string  $q$  of length  $n$ . Each character in  $q$  is either “+” or “?” (without quotes). Here, “+” means one call of “add”, and “?” means one call of “query”. These operations are performed according to the order in  $q$ .

### Output

Output a single integer  $t$ : the minimum total time in seconds it will take for  $S$  to process all queries after possibly adding some calls of “rebuild”.

### Examples

standard input	standard output
6 5 ++??+?	6
6 8 ++??+?	7
5 1 +++++	0

### Explanations

In the first example, the most optimal way is to place “rebuild” before the first “?”.



In the second example, no placing of “**rebuild**” operations can decrease the total time.

In the third example, you also cannot decrease the total time because there are no “**query**” operations at all.