

ODIX

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Manuel is still obsessed with the acronym **ODI**. Recently, he has been playing with an L -length string S consisting of those three letters and the letter **X**. He wants to find the longest substring of S that has exactly n **O**'s, m **D**'s, and k **I**'s.

For example, if $S = \text{IDOOXODIXD}$, $n = 2$, $m = 1$, $k = 1$, the longest substring containing *two* **O**'s, *one* **D**, and *one* **I** is **OXODIX**, which starts at position 4 and ends at position 9 of S .

Input

The first line of input consists of integer L ($3 \leq L \leq 3 \times 10^5$), the length of string S .

The second line contains S , a string of L letters. Each letter is either an **O**, **D**, **I**, or **X**.

The third line consists of three positive integers n , m , and k ($n + m + k \leq L$), the exact number of occurrences of **O**s, **D**s, and **I**s, respectively, that must appear in the longest substring you must find.

Output

In a single line, print out a single integer, the length of the longest substring of S with exactly n **O**'s, m **D**'s, and k **I**'s. If there is no substring of S that satisfies this condition, print -1.

Scoring

This problem is divided into the following subtasks.

Subtask	Points	Additional Restrictions
0	0	Example test cases
1	11	$L = 3$, $n = m = k = 1$
2	16	$L \leq 3000$
3	30	There are no X s in string S
4	43	No additional restrictions

Examples

standard input	standard output
10 IDOOXODIXD 2 1 1	6
7 ODIXIOI 2 1 1	-1
16 IDIOXXXDDDIODXX 2 3 1	11