Task: CTML
CaTML

AACPP WiSe 2025/26 Round 1 Memory: 512MiB 2025.10.22 – 2025.11.05

Dexter is plotting a revenge on Mat for not giving him enough snacks. When Mat left the room, Dexter jumped on his desk and wanted to do something mischievous on his computer.

On the screen there is some code in CaTML. Dexter doesn't understand much of it, but he gets that there are opening and closing tags. For simplicity, we represent them as a sequence of '<' and '>' symbols, respectively. *Valid* CaTML programs are defined as a sequence of opening and closing tags such that each opening tag has a matching closing tag. Equivalently, valid CaTML programs are:

- the program <>;
- programs < P >, where P is a valid CaTML program;
- programs P_1P_2, where P_1, P_2 are valid CaTML programs.

Dexter wants to ruin the CaTML code Mat was writing – currently of length n and not necessarily valid – by jumping on the keyboard and inserting random characters in $k-1$ places, thus breaking up the code into k non-empty subsequences. To achieve maximum chaos, he wants the subsequences to be as invalid as possible. More precisely, he wants to *minimise the sum, over all created subsequences, of valid CaTML programs that are contained within*. Help Dexter and find that sum.

Input

he first line of input contains two integers n, k, the length of the existing sequence and the number of subsequences to create, respectively.

The second line contains a word of length n containing only characters '<' and '>'.

Output

Your program should write one line to the standard output – the minimal possible sum of correct programs contained within k separated subsequences.

Example

For the input:

```
15 2
<><><><><><><>
```

the correct output is:

```
6
```

while the input:

```
15 3
<><><><><><><>
```

the correct output is:

```
3
```

Explanation: In the first test, an optimal split is, for example:

<><><><><><><> ⟶ <><><>< | ><><><>

The first subsequence contains two valid CaTML programs:

- <u><></u><><>,
- <><><u><></u>,

The second subsequence contains four:

- >><><<>>,
- >><><><>>,
- >><><<><>>,
- >><><><<>>.

The sum equals 6.

In the second example, an optimal division could be:

<><><><><><><> ⟶ <><>< | ><><><>< | >>

Additional examples

The following initial tests are also available:

- `0c` – $n = k = 300$, all opening tags;
- `0d` – $n = 4\,000$, $k = 2\,000$, repeating sequence of <>;
- `0e` – $n = 100\,000$, $k = 30$, first $50\,000$ opening tags, then $50\,000$ closing tags;
- `0f` – $n = k = 100\,000$, first $50\,000$ closing tags, then $50\,000$ opening tags;

Limits

Your solution will be evaluated on a number of hidden test cases divided into groups. Points for a group are awarded if and only if the submission returns the correct answer for each of the tests in the group within the allotted time limit. These groups are organised into subtasks with the following limits and points awarded.

Subtask	Limits	Points
1.	$1 \leq n, k \leq 300$	2
2.	$1 \leq n, k \leq 4\,000$	2
3.	$1 \leq n \leq 100\,000$, $1 \leq k \leq 30$	2
4.	$1 \leq n, k \leq 100\,000$	4