Problem E
Evil Judges
Time Limit: 1 Second, Memory Limit: 2G

The evil judges preparing the ICPC problem sets are tired of seeing the talented contestants AC their problems and AK ("all kill") their problem sets. They have grown so tired of this that they've started to dislike any strings where `AC` or `AK` appear as subsequences. The judges just found a string s consisting of only the characters `A`, `C`, and `K` and are determined to destroy these subsequences!

In one operation, the judges are able to swap two adjacent characters in the string s. To be more precise, the judges may choose an index i ($1 \leq i < |s|$) and swap s_i and s_{i+1}. The judges are very busy and don't have much time, so they can only do this operation up to M times (independently choosing an index i each time).

The judges' goal is to minimize the number of subsequences (**possibly non-contiguous**) that are `AC` or `AK`. Help them calculate the number of `AC` and `AK` subsequences that remain within s after the judges perform an optimal sequence of up to M swap operations.

Input

The first line of input contains the string s ($1 \leq |s| \leq 2 \cdot 10^5$). The string consists only of the characters `A`, `C`, and `K`.

The second line contains an integer M ($0 \leq M \leq 2 \cdot 10^9$), the maximum number of swaps the judges can perform.

Output

Print the minimum total number of `AC` and `AK` subsequences that remain in s after an optimal sequence of at most M swap operations.

Explanation of Sample Input 1

In Sample Input 1, the string s initially has seven different `AC` subsequences and four different `AK` subsequences. One optimal choice of five swaps yields the new string `ACCCAAKA` which only has three `AC` subsequences and three `AK` subsequences left.

Sample Input 1	Sample Output 1
ACAACCAK 5	6

Sample Input 2	Sample Output 2
CCKKAKCKCK 1000000000	0

Sample Input 3	Sample Output 3
AAAAAAAAACCCCCCCCC 13	68