Apple Sequences

There is a string of size **n** containing only '**A**' and '**O**'. '**A**' stands for Apple, and '**O**' stands for Orange. We have *m* number of spells, each spell allows us to convert an orange into an apple.

Find the longest sequence of apples you can make, given a string and the value of **m**.

Example 1:

```
Input:
N = 5
M = 1
arr[] = 'AAOAO'
Output: 4
Explanation: Changing the orange at
3rd position into an apple gives
us the maximum possible answer.
```

Example 2:

```
Input:
N = 5
M = 1
arr = 'AOOAO'
Output: 2
Explanation: Changing any orange into
an apple will give us a sequence
of length 2.
```

Your Task:

You don't need to read input or print anything. Your task is to complete the function **appleSequence()** which takes the array in the form of a string, its size n, and an integer m as input parameters and returns the largest apple sequences after converting m oranges.

Expected Time Complexity: O(n)
Expected Auxiliary Space: O(1)

Constraints:
$1 <= m <= n <= 10^6$