

Balanced Workshops

time limit per test: 3 seconds (3 seconds for Python)
memory limit per test: 500 megabytes
input: standard input
output: standard output
points: 100 · problem code: 1008stackqueue1

Brendon is trying to write the perfect workshop questions to keep you motivated.

A workshop question normally has a bunch of individual moments of learning. Brendon has broadly separated the moments into one of two categories:

- "Idea" moments - students begin to get a concept
- "wow..." moments - an idea is brought to its completion, and the concept is learned.

Any question Brendon writes can have multiple such moments. Brendon's workshop will contain two questions in sequence (essentially we are concatenating the moments of two question sequences).

A workshop is balanced if every wow... moment is matched with a unique Idea moment that occurred before it, and vice-versa.

In other words, every idea is followed up with a second "wow..." moment that ensures the student learns the concept, and there is no "wow..." moment that isn't being used for this purpose.

Brendon has a few questions to choose from - can you tell him how many ordered pairs of questions can be used to create a balanced workshop?

In order to pass the tests, your solution should have a time complexity of $O(a)$ - the total number of moments between all questions.

Input

Input will begin with two integers, the number of questions n ($1 \leq n \leq 3 \times 10^5$), and the total number of moments in these questions combined a ($1 \leq a \leq 3 \times 10^5$).

The next line will contain n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq a$), representing the number of moments that question has. You can be guaranteed that the sum of all a_i will always equal the a provided on the previous line.

a lines will follow, and will either be the word "Idea" or the word "wow...". These are the n questions read out sequentially.

Output

Output a single integer, the number of ordered pairs of questions which can be combined to make a balanced workshop.

Example 1

Input

```
3 4
1 2 1
wow...
Idea
wow...
Idea
```

Output

```
1
```

The input represents three questions. The first is a single "wow..." moment. The second is an "Idea", followed by a "wow..." moment. The third is a single "Idea" moment.

The only way to build a balanced workshop from 2 questions is to start with question 3, and end with question 1. Question 2 is balanced by itself, but when paired with question 1 or question 3, it would not be balanced.

While question 1 followed by question 3 does have an equal number of both moments, but it is not balanced as the wow... moment occurs before the Idea moment.

Example 2

Input

```
2 4
2 2
Idea
wow...
Idea
wow...
```

Output

```
2
```

This input represents two questions, which both have an "Idea" moment followed by a "wow...". Either ordering of the two questions would make for a balanced workshop, since every "Idea" moment can be matched to a "wow..." moment.

Example 3

Input

```
5 11
3 2 3 1 2
Idea
Idea
wow...
Idea
wow...
Idea
wow...
Idea
wow...
Idea
wow...
```

Output

```
4
```

There are 5 questions in this input:

- Q1: Idea Idea wow...
- Q2: Idea wow...
- Q3: Idea wow... Idea
- Q4: wow...
- Q5: Idea wow...

The possible balanced workshops are: Q1+Q4, Q3+Q4, Q2+Q5 and Q5+Q2.