

Task: DDL

Two long candy sticks



XXV OI, Stage III, Day two. Source file `ddl.*` Available memory: 256 MB.

12.04.2018

Byteasar runs a candy store in Byteburg. All children love all things sweet, but little Byteburgers have a favorite: the strawberry-vanilla candy sticks. These consist of segments of alternating flavors (vanilla and strawberry), which, thanks to Byteasar's secret recipe, may have arbitrary length.

Bitie and Bytie visit the store frequently, each time buying one famous candy stick per head. But unless Byteasar sells them two pieces that are *flavor equivalent*, i.e., have the same total length of the strawberry flavored segments *and* the same total length of the vanilla flavored segments, the boys immediately start arguing which piece is better. The boys care little for actual segment arrangements — the total amount of each flavor is all that matters to them.

Bitie and Bytie are entering the store right now, just as Byteasar has only two long candy sticks available. He intends to break the two sticks (simultaneously, one in each hand!) so that Bitie gets a piece of the first one whereas Bytie a piece of the second and the pieces are flavor equivalent. Byteasar can give each boy an arbitrary contiguous (i.e., unbroken) fragment of the stick intended for him. To this end, he may break the sticks multiple times at any positions, not necessarily at segment boundaries.

Help Byteasar determine how he should break each candy stick, so that he sells the boys two pieces, one per original stick, so that so that the pieces are flavor equivalent and their (common) length is maximized.

Input

The two long candy sticks are described on the standard input, one after another, each in the following format. The first line of the description contains a single positive integer m , which specifies the number of the candy stick's segments. Then m lines follow, the i -th of which contains a single character t_i and a positive integer a_i , separated by a single space. The character t_i specifies the segment's flavor: T for strawberry (*truskawka* in Polish), and W for vanilla (*wanilia* in Polish). The number a_i specifies the segment's length (in centimeters). You may assume that the flavors of two successive segments are different.

Output

Your program should print a single integer (in a single line) to the standard output: the maximum (common) length of flavor equivalent pieces that can be broken off, one from each stick.

Example

For the input data:

```
3
T 5
W 7
T 4
3
W 6
T 6
W 6
```

the correct result is:

```
13
```

Explanation for the example: Byteasar can sell the boys pieces that are each 7 cm long vanilla and 6 cm long strawberry flavored. From the first candy stick he should discard 3 cm of strawberry flavor, arbitrarily distributed over the ends, whereas from the second one he should discard 5 cm of vanilla flavor, again arbitrarily distributed over the ends.

Sample grading tests:

1ocen: The first segment of the first stick is 30 cm of vanilla, and its successive segments are 1 cm long, whereas the second stick consists of alternating strawberry and vanilla segments of lengths 1 cm and 2 cm respectively. The first candy stick consists of 5 segments, while the second of 8. The correct answer is 8.

2ocen: Each candy stick has 10 segments, strawberry and vanilla alternately. Successive segment lengths are $(1, 2, 3, \dots, 10)$ for the first and $(10, 9, 8, 7, 6, 1, 2, 3, 4, 5)$ for the second stick. The correct answer is 50.

3ocen: Each candy stick has 1000 segments, the first one starting with strawberry and the second with vanilla, and every segment in either stick is 1 cm long. The correct answer is 1000.

Grading

The set of tests consists of the following subsets. Within each subset, there may be several unit tests.

In the table below, n_{\max} denotes the maximum number of segments of a candy stick, whereas m_{\max} – the maximum length of a stick. You may assume that $n_{\max} \leq 10^9$, $m_{\max} \leq 1000$ hold in every unit test.

Subset	Property	Score
1	$n_{\max} \leq 150$	8
2	$n_{\max} \leq 3000$	12
3	$n_{\max} \leq 500\,000, m_{\max} \leq 150$	12
4	$m_{\max} \leq 150$	8
5	$n_{\max} \leq 500\,000, m_{\max} \leq 300$	15
6	$m_{\max} \leq 300$	9
7	$n_{\max} \leq 500\,000$	21
8	no further constraints	15