Problem G. Global Elephant Market

Input file:	`stdin`
Output file:	`stdout`
Time limit:	3 seconds
Memory limit:	256 megabytes

— Buy an elephant!
— Why would I want an elephant?
— Everyone asks, "What I need it for", just come and buy an elephant.
— Get off me!
— I will, but first you gotta buy an elephant!

This is an interactive problem.

Welcome to the world's largest elephant market. As you are a newbie to the market, your current goal is to learn how to analyze the current state of the market. You should be able to answer how much money you can make at any given moment, assuming that you have enough initial money to be able to make all the deals you wish. You will not perform any real trade.

The elephant market is based on an auction market model where a buyer bids a specific price for an elephant and a seller asks a specific price for an elephant. The prices are updated constantly as traders change their mind.

We assume that, given the current market situation, you can buy a few elephants but you must immediately sell them. You are trading just hand-to-mouth as you are not interested in keeping elephants even for a minute. So, you have to buy and sell the same number of elephants in total.

At the opening bell of the elephant market, no one wants to buy or sell elephants. Elephant offers are placed or withdrawn sequentially, listing what number of elephants one can sell or what number of elephants one can buy at specified prices. To keep things simple, you will only see a change of the number of elephants for buying or selling at the specified price. For example, "`buy 10 100`" means that you can now sell 10 more elephants for 100 piastres each and "`sell -3 99`" means that you can now buy 3 less elephants for 99 piastres each.

Write a program to determine the maximum possible profit (amount of money) you can make on the market after each offer to buy or sell elephants is placed. Note that as you do not actually perform any real trade, you must assume that after each new offer is placed, all the previous offers are still valid too.

Input

Your program will receive market offers sequentially in the following format. Every line of input corresponds to one offer and starts with the offer type (one of strings "`buy`", "`sell`" or "`end`") meaning an offer to buy some elephants, an offer to sell some elephants and the end of the trading session respectively. Offer "`end`" contains no additional parameters, and your program must exit immediately after it. Other offers contain two integers D and P separated by a single space as additional parameters describing a change D of the number of elephants on the market with appropriate order type and price P ($-10^6 \le D \le 10^6$, $1 \le P \le 10^9$). The total number of offers will never exceed 10^5.

It is guaranteed that both the total buy price of all elephants currently available on the market and the total sell price of all elephants currently available on the market will not exceed 2^{62}, and the total number of elephants for buying or selling with any price will never become negative.

Output

After each "`buy`" or "`sell`" offer, you must output a single integer on a separate line: the maximum

possible profit in piastres you can make after this offer is placed on the market. Do not forget to print end-of-line characters and to flush output buffers after each output.

Examples

stdin	stdout
buy 10 100 sell 4 98 buy -7 100 buy 2 99 sell 1 97 end	0 8 6 7 9