Problem A. John's Fantasy

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Implementing Stack is necessary for this lab.

John is a boss at a trading firm.

He has full authority to hire new employees, fire employees. So he maintains a list of employees which is sorted in increasing order of their joining date.

The employees also demands an increment to enjoy the **Happy Weekend**. Considering their requests, he decides to increment the salary of first p employees only.

Here **first** p means the employees who have earlier joining date. Initially the salary of employee is decided when he joins the firm.

In addition he will only fire the employee who have the latest joining date since John follow the saying that "Old is Gold".

Initially the firm has no employees. Since John has written the commands in a hurry he can ask you to fire a employee even when the firm has no employees working in it.

Since John has a meeting scheduled this weekend he assign you a list of commands and asked you to print the salary of each employee after these commands are executed on the employee's list.

The commands are of 3 types :

- **1 s** — a new employee joins the firm with initial salary s.
- **2** — a employee is fired from the firm
- **3 p i** —– salary of the first p employee's is increased by amount i.

Input

The first line of input contains n, the number of commands John asked you to perform.

Each of the next n lines contains a single command of one of three types.

- **1 s** — a new employee joins the firm with initial salary s.
- **2** — a employee is fired from the firm
- **3 p i** —– salary of the first p employee's is increased by amount i.

$1 \leq n \leq 10^5$

$1 \leq s, p, i \leq 10^5$

$p \leq \mathit{current\ size\ of\ list}$ for every command from 1 to n.

Note: Look closely at the constraints, time limit, and think about the time complexity of your solution.

Output

The only line of output contains the salary of employee's in order of their joining time. Else print -1 if there are no employee's left in the firm.

Examples

standard input	standard output

5
1 1
1 2
1 3
1 4
1 5
	1 2 3 4 5

3
1 1
1 2
3 2 4
	5 6

5
1 1
3 1 5
2
1 5
3 1 1
	6

5
1 1
1 9
2
2
1 10
	10

8
1 3
1 2
1 7
3 2 8
1 1
1 3
3 5 1
2
	12 11 8 2

5
1 2
1 4
2
2
2
	-1

9
1 3
1 2
2
1 4
3 2 9
2
3 1 1
2
2
	-1

Problem B. Special Friend 2

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Shivansh and his *special* friend are again meeting after celebrating the happy weekend. He will share the problems of upcoming lab to his special friend is able to correctly guess the output of the game.

This time they are playing a game of zero increment. Initially the number is zero.

The game consists of 3 commands which shivansh will ask her to perform.

Let's take at any point of game the value of the number is x then the operations are :

- **adds** – add 1 to x
- **for** n – for loop
- **end** —– every command between "for n" and corresponding "end" is executed n times

Every "for n" is matched with "end", thus the input is always valid.

——

After the execution of these commands, she writes the value of x on a paper.

But here is a **catch** that paper can hold number whose value is in between 0 and $2^{31} - 1$. So if the number is not in this range,she writes "Time wasted" on the paper else writes the corresponding number.

For his special friend the instructions are too much to keep track of. So she has asked you to write a program for it.

(See sample test case for reference).

Input

The first line of input contains a single integer q —– the number of lines in input.

Each of the next q lines contains a single command of one of three types.

- **adds** – add 1 to x
- **for** n – for loop
- **end** —– every command between "for n" and corresponding "end" is executed n times

$1 \leq q \leq 10^5$

$1 \leq n \leq 100$

Output

If the result overflows the range then output "Time wasted", otherwise print the resulting value of the number.

Examples

standard input	standard output

9
add
for 5
end
for 5
for 5
add
end
add
end
	31

2
for 62
end
	0

11
for 100
for 100
for 100
for 100
for 100
add
end
end
end
end
end
	Time wasted

5 add add add add add	5

11
for 100
for 100
for 100
for 100
for 100
end
add
end
end
end
end
	100000000

3
for 100
add
end
	100