Problem F: Queue at '63 South

Filename: `queue`

Time limit: `2 seconds`

At the legendary ‘63 South cafeteria at the University of Central Florida campus, students attending the ICPC Competitive Programming Training Camp follow a special queuing system. To encourage camaraderie and group decision-making, students form groups before entering the queue. Each group joins the back of the queue together.

However, when it’s time to eat, students are called one at a time from the front of the queue—regardless of what group they came with.

You’re given a sequence of events, each of which is either:

- A group of students joining the back of the queue, or
- A single student leaving from the front.

You need to process a sequence of these events and output the number of students currently in the queue after each event.

The Problem

Process a list of queue operations and output the number of students in the queue after each one.

The Input

Input starts with a single integer **c**, the number of test cases.

The first line of each test case contains an integer **q** — the number of events. ($1 \le \mathbf{q} \le 10^5$)

Each of the next q lines is either:

- "E x" — a group of **x** students enters the queue, or
- "D" — one student departs from the front of the queue.

It is guaranteed that a "D" operation will never occur when the queue is empty.

The Output

Output a single line with **q** space-separated integers — after each event, print a single integer: the current number of students in the queue.

Input Bounds and Corresponding Credit

30 Points	70 Points
• 1 ≤ **c** ≤ 5 • 1 ≤ **q** ≤ 100 • For all "E x" operations: 1 ≤ **x** ≤ 10	• 1 ≤ **c** ≤ 10 • 1 ≤ **q** ≤ 10^5 • For all "E x" operations: 1 ≤ **x** ≤ 10^9

Samples

Input	Output
2 6 E 3 E 2 D E 1 D D 4 E 2 E 1 D D	3 5 4 5 4 3 2 3 2 1

Sample Explanation:

Explanation of Subtest #1:

- E 3 → 3 students enter: queue = [3] → total = 3
- E 2 → 2 students enter: queue = [3, 2] → total = 5
- D → 1 student from front group (3) leaves → group becomes [2] → total = 4
- E 1 → 1 student enters: queue = [2, 2, 1] → total = 5
- D → 1 student from front group (2) leaves → group becomes [1] → total = 4
- D → 1 student from front group (1) leaves → group is removed → total = 3

Explanation of Subtest #2:

- E 2 → Group of 2 enters: queue = [2] → total = 2
- E 1 → Group of 1 enters: queue = [2, 1] → total = 3
- D → 1 leaves from group 2 → group becomes [1] → total = 2
- D → 1 leaves from group 2 → group removed → total = 1