Tasks Planner

Create a program that helps you organize your daily tasks. First, you are going to **receive the hours each task takes** on a **single line, separated by space**, in the following **format**:

"{task$_1$} {task$_2$} {task$_3$}… {task$_n$}"

Each task takes **from 1 to 5 hours**. If its time is **set to 0** – it is **completed**. If its time is **set to a negative number** – the task is **dropped**.

Then you will start receiving **commands** until you read the "**End**" message. There are **six** possible commands:

- **"Complete {index}"**
  - Find **the task** on this **index** in your collection and complete it, if the **index exists**.
- **"Change {index} {time}"**
  - **Replace** the time needed of the **task** on the given index **with the time given,** if the **index exists**.
- **"Drop {index}"**
  - **Drop** the task on the given **index, setting its hour to -1,** if the **index exists**.
- **"Count Completed"**
  - Print the number of **completed** tasks.
- **"Count Incomplete"**
  - Print the number of **incomplete** tasks (this **doesn't include** the **dropped** tasks).
- **"Count Dropped"**
  - Print the number of **dropped** tasks (this **doesn't include** the **incomplete** tasks).

In the end, print the **incomplete tasks** on a **single line,** separated by a **single space** in the following format:

"{task$_1$} {task$_2$} {task$_3$}… {task$_n$}"

Input

- On the **1st line** you are going to receive the **time of each task**, separated by a single space.
- On the next **lines**, until the **"End"** command is received, you will be receiving commands.

Output

- Print the tasks in the **format described above**.

Examples

Input	Output
1 -1 2 3 4 5 Complete 4 Change 0 4 Drop 3 Count Dropped End	2 4 2 5
Comments	

First, we receive the command "**Complete 4**" and we to complete the task on index 4. After this command, the task collection looks like this:

1 -1 2 3 0 5

Afterwards, we receive the "**Change 0 4**" command and we need to change the time of the task on index 0. The collection looks like this now:

4 -1 2 3 0 5

After, we receive the "**Drop 3**" command, which means we need to drop the task on index 3. The collection looks like this:

4 -1 2 -1 0 5

Then, we receive the "**Count Dropped**" command. The result is 2 as we have only 2 dropped tasks.

In the end, we print all of the **incomplete** tasks. This is the result collection:

4 2 5

1 2 3 4 5 4 0 3 2 1 Complete 0 Complete 1 Complete 2 Drop 3 Change 4 1 Count Completed End	4 1 4 3 2 1