# Problem 3 – Byte Party

You will be given an integer number N and on each of the next N lines - an 8-bit unsigned integer. On the next lines you will be given a series of commands, one of the following:

* **"-1 [position]"** – Upon receiving this command you should flip the bits at the specified position in all numbers you received. Flipping a bit means turning its value from 1 to 0 or the other way around.
* **"0 [position]"** – upon receiving this command you should unset the bits at the specified position for all numbers, i.e. turn all bits to 0 regardless of their current value.
* **"1 [position]"** – upon receiving this command you should set the bits at the specified position for all numbers, i.e. turn all bits to 1 regardless of their current value.
* **"party over"** – when you receive this command print back the numbers after all changes have been made; each number stays on a separate line.

## Input

* The input data should be read from the console.
* The first input line holds **the number N – the count of integers** you'll receive.
* On each of the next **N lines** you'll receive an integer number. Input ends with **the string "party over".**
* The input data will always be valid and in the format described. There is no need to check it explicitly.

## Output

* The output data should be printed on the console.
* You should **print N lines, each containing a number** – the numbers after all manipulations.

## Constraints

* All **input numbers** are in the range [0 … 255].
* **[position]** will be between [0 … 7].
* Allowed working time for your program: 0.1 seconds. Allowed memory: 16 MB.

## Examples

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | | | | | | | | | | | **Output** | | | | |
| 3  44  106  12  -1 0  0 1  1 2  party over | | | | | | | | | | | | 45  109  13 | | | | |
| 7 | | 6 | | | 5 | | 4 | 3 | | 2 | | | 1 | 0 | |  |
| 0 | | | 0 | 1 | | 0 | | | 1 | | 1 | 0 | | 0 | 44 | |
| 0 | | | 1 | 1 | | 0 | | | 1 | | 0 | 1 | | 0 | 106 | |
| 0 | | | 0 | 0 | | 0 | | | 1 | | 1 | 0 | | 0 | 12 | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 6 | 5 | | 4 | | 3 | | 2 | | 1 | | 0 | |  |
| 0 | 0 | | 1 | | 0 | | 1 | | 1 | | 0 | | 1 | 45 |
| 0 | 1 | | 1 | | 0 | | 1 | | 1 | | 0 | | 1 | 109 |
| 0 | 0 | | 0 | | 0 | | 1 | | 1 | | 0 | | 1 | 13 |