

# Killing Monsters

| Problem Code: **MONSTER**

---

Chef is playing a game named "Killing monsters". In this game, at the beginning, there are  $n$  monsters arranged in a row (numbered 0 through  $n-1$  from left to right). The  $i$ -th monster initially has  $h_i$  health points. A monster dies if its health points become  $\leq 0$ .

Chef is going to perform  $q$  actions. For each action, Chef chooses two numbers  $x, y$  and shoots all living monsters whose index  $k$  satisfies  $k \& x = k$  ( $\&$  denotes the bitwise AND operator). Each monster he shoots loses  $y$  health points.

After each action, please help Chef determine the number of living monsters.

---

## Input

- The first line of the input contains a single integer  $n$  denoting the number of monsters.
  - The second line contains  $n$  space-separated integers  $h_0, h_1, \dots, h_{n-1}$  denoting the initial health of monsters.
  - The third line contains a single integer  $q$  denoting the number of actions.
  - $q$  lines follow. Each of these lines contains two space-separated integers  $x$  and  $y$  describing one action.
- 

## Output

For each action, print a single line containing one integer — the number of living monsters afterwards.

---

## Constraints

- $1 \leq n \leq 2^{17}$
  - $1 \leq h_i \leq 10^9$  for each valid  $i$
  - $1 \leq q \leq 2^{18}$
  - $0 \leq x \leq 10^9$
  - $1 \leq y \leq 10^9$
- 

## Subtasks

**Subtask #1 (10 points):**

- $1 \leq n \leq 2^{10}$
- $1 \leq q \leq 2^{11}$

**Subtask #2 (20 points):**

- $h_i = i$  for each valid  $i$
- $y = 1$  for each action

**Subtask #3 (70 points):** original constraints

## Example

**Input :**

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

**Output :**

4  
4  
2  
2  
1

Author: 7★ [chemthan \(/users/chemthan\)](/users/chemthan)

Date Added: 4-11-2017

Time Limit: 3 secs

Source Limit: 50000 Bytes

Languages: ADA, ASM, BASH, BF, C, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 6.3, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK, ICON, JAVA, JS, kotlin, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PAS fpc, PAS gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.5, RUBY, rust, SCALA, SCM chicken, SCM guile, SCM qobi, ST, swift, TCL, TEXT, WSPC