

## Problem A

---

### A+B

---

You are given two integers  $A$  and  $B$ . Print the calculation result of  $A + B$ .

### Input

---

Input is given from standard input in the following format:

$A$   $B$

- $1 \leq A, B \leq 10^9$

### Output

---

You have to output the calculation result of  $A + B$  to the standard output.

### Sample Input 1

---

3 5

### Sample Output 1

---

8

### Sample Input 2

---

10000 100

### Sample Output 2

---

10100

## Problem A

---

### A+B

---

2つの整数  $A, B$  が与えられます。 $A + B$  の計算結果を出力してください。

### 入力

---

入力は以下の形式で標準入力から与えられます。

```
A B
```

### 出力

---

$A + B$  の計算結果を標準出力に出力してください。

### 制約

---

- $1 \leq A, B \leq 10^9$

### 入力例 1

---

```
3 5
```

### 出力例 1

---

```
8
```

### 入力例 2

---

```
10000 100
```

### 出力例 2

---

```
10100
```

## Problem B

---

### 回文判定

---

長さ  $N$  の文字列  $S$  が与えられます。文字列は奇数長であることが保証されます。

与えられる文字列  $S$  が回文であるか判定してください。

### 入力

---

$N$   
 $S$

- $1 \leq N \leq 100000$
- $N$  は奇数であることが保証される
- $|S| = N$
- 与えられる文字列は英小文字のみからなる

### 出力

---

与えられる文字列が回文であれば **Yes** を、そうでなければ **No** を一行で出力してください。出力の末尾で改行してください。

### 入出力例

---

#### 入力例 1

---

apa

#### 出力例 1

---

Yes

#### 入力例 2

---

bed

#### 出力例 2

---

No



## Problem C1

---

### Static Range Sum

---

#### Problem Statement

---

You are given a non-negative integer sequence  $A = (a_0, a_1, \dots, a_{N-1})$  with the length  $N$ . Process the following  $Q$  queries in order:

- You are given integers  $l_i$  and  $r_i$ . Print  $\sum_{k=l_i}^{r_i-1} a_k$ .

#### Note

---

The difference between C1 and C2 is only constraints of  $N$  and  $Q$ .

#### Input

---

```
 $N$   $Q$   
 $a_0$   $a_1$  ...  $a_{N-1}$   
 $l_1$   $r_1$   
 $\vdots$   
 $l_Q$   $r_Q$ 
```

- All inputs are integers.
- $1 \leq N \leq 1\,000$
- $1 \leq Q \leq 1\,000$
- $-10^9 \leq a_i \leq 10^9$
- $1 \leq l_i < r_i \leq N$

#### Sample

---

#### Sample Input

---

```
5 5  
1 10 100 1000 10000  
2 3  
0 3  
2 5  
3 4  
0 5
```

#### Sample Output

---

International Collegiate Programming Contest  
Practice Contest for ICPC YYYY Domestic Contest, YYYY-MM-DD

```
100
111
11100
1000
11111
```

## Problem C1

---

### Static Range Sum

---

#### 問題文

---

長さ  $N$  の非負整数列  $A = (a_0, a_1, \dots, a_{N-1})$  が与えられます。以下で説明されるクエリを順に  $Q$  回処理してください。

- 整数  $l_i$  と  $r_i$  が与えられるので、 $\sum_{k=l_i}^{r_i-1} a_k$  を出力する。

#### 注意

---

問題 C1 と C2 では、 $N$  と  $Q$  の制約のみが異なります。

#### 入力

---

```
N Q
a0 a1 ... a_{N-1}
l1 r1
⋮
l_Q r_Q
```

- 入力は全て整数で与えられる
- $1 \leq N \leq 1,000$
- $1 \leq Q \leq 1,000$
- $-10^9 \leq a_i \leq 10^9$
- $1 \leq l_i < r_i \leq N$

#### 入出力例

---

#### 入力例

---

```
5 5
1 10 100 1000 10000
2 3
0 3
2 5
3 4
0 5
```

#### 出力例

---

International Collegiate Programming Contest  
Practice Contest for ICPC YYYY Domestic Contest, YYYY-MM-DD

```
100
111
11100
1000
11111
```



## Problem C2

---

### Static Range Sum

---

#### Problem Statement

---

You are given a non-negative integer sequence  $A = (a_0, a_1, \dots, a_{N-1})$  with the length  $N$ . Process the following  $Q$  queries in order:

- You are given integers  $l_i$  and  $r_i$ . Print  $\sum_{k=l_i}^{r_i-1} a_k$ .

#### Note

---

The difference between C1 and C2 is only constraints of  $N$  and  $Q$ .

#### Input

---

```
 $N$   $Q$   
 $a_0$   $a_1$  ...  $a_{N-1}$   
 $l_1$   $r_1$   
 $\vdots$   
 $l_Q$   $r_Q$ 
```

- All inputs are integers.
- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $-10^9 \leq a_i \leq 10^9$
- $1 \leq l_i < r_i \leq N$

#### Sample

---

#### Sample Input

---

```
5 5  
1 10 100 1000 10000  
2 3  
0 3  
2 5  
3 4  
0 5
```

#### Sample Output

---

International Collegiate Programming Contest  
Practice Contest for ICPC YYYY Domestic Contest, YYYY-MM-DD

```
100
111
11100
1000
11111
```

## Problem C2

### Static Range Sum

#### 問題文

長さ  $N$  の非負整数列  $A = (a_0, a_1, \dots, a_{N-1})$  が与えられます。以下で説明されるクエリを順に  $Q$  回処理してください。

- 整数  $l_i$  と  $r_i$  が与えられるので、 $\sum_{k=l_i}^{r_i-1} a_k$  を出力する。

#### 注意

問題 C1 と C2 では、 $N$  と  $Q$  の制約のみが異なります。

#### 入力

```
N Q
a0 a1 ... a_{N-1}
l1 r1
⋮
lQ rQ
```

- 入力は全て整数で与えられる
- $1 \leq N \leq 100,000$
- $1 \leq Q \leq 100,000$
- $-10^9 \leq a_i \leq 10^9$
- $1 \leq l_i < r_i \leq N$

#### 入出力例

#### 入力例

```
5 5
1 10 100 1000 10000
2 3
0 3
2 5
3 4
0 5
```

#### 出力例

International Collegiate Programming Contest  
Practice Contest for ICPC YYYY Domestic Contest, YYYY-MM-DD

```
100
111
11100
1000
11111
```

## Problem D

---

### A+B

---

2つの整数  $A, B$  が与えられます。 $A + B$  を出力してください。

$T$  個のテストケースが与えられるので、 $T$  行出力してください。

### Input

---

```
T
A1 B1
A2 B2
...
AT BT
```

- 入力は全て整数で与えられる
- $1 \leq T \leq 100,000$
- $1 \leq A_i, B_i \leq 10^9$

### Output

---

$T$  行出力してください。

$i$  行目には  $A_i + B_i$  の計算結果を出力してください。

### Sample Input

---

```
3
3 9
100 101
1 10
```

### Sample Output

---

```
12
201
11
```

## H: Many Kinds of Apples

---

### Problem Statement

---

Apple Farmer Mon has two kinds of tasks: "harvest apples" and "ship apples".

There are  $N$  different species of apples, and  $N$  distinguishable boxes. Apples are labeled by the species, and boxes are also labeled, from 1 to  $N$ . The  $i$ -th species of apples are stored in the  $i$ -th box.

For each  $i$ , the  $i$ -th box can store at most  $c_i$  apples, and it is initially empty (no apple exists).

Mon receives  $Q$  instructions from his boss Kukui, and Mon completely follows in order. Each instruction is either of two types below.

- "harvest apples": put  $d$   $x$ -th apples into the  $x$ -th box.
- "ship apples": take  $d$   $x$ -th apples out from the  $x$ -th box.

However, not all instructions are possible to carry out. Now we call an instruction which meets either of following conditions "impossible instruction":

- When Mon harvest apples, the amount of apples exceeds the capacity of that box.
- When Mon tries to ship apples, there are not enough apples to ship.

Your task is to detect the instruction which is impossible to carry out.

### Input

---

Input is given in the following format.

```
N
c1 c2 ... cN
Q
t1 x1 d1
t2 x2 d2
⋮
tQ xQ dQ
```

In line 1, you are given the integer  $N$ , which indicates the number of species of apples.

In line 2, given  $c_i$  ( $1 \leq i \leq N$ ) separated by whitespaces.  $c_i$  indicates the capacity of the  $i$ -th box.

In line 3, given  $Q$ , which indicates the number of instructions. Instructions are given

successive  $Q$  lines.  $t_i x_i d_i$  means what kind of instruction, which apple Mon handles in this instruction, how many apples Mon handles, respectively. If  $t_i$  is equal to 1, it means Mon does the task of "harvest apples", else if  $t_i$  is equal to 2, it means Mon does the task of "ship apples".

## Constraints

---

All input values are integers, and satisfy the following constraints.

- $1 \leq N \leq 1,000$
- $1 \leq c_i \leq 100,000$  ( $1 \leq i \leq N$ )
- $1 \leq Q \leq 100,000$
- $t_i \in \{1, 2\}$  ( $1 \leq i \leq Q$ )
- $1 \leq x_i \leq N$  ( $1 \leq i \leq Q$ )
- $1 \leq d_i \leq 100,000$  ( $1 \leq i \leq Q$ )

## Output

---

If there is "impossible instruction", output the index of the apples which have something to do with the first "impossible instruction". Otherwise, output 0.

## Sample Input 1

---

```
2
3 3
4
1 1 2
1 2 3
2 1 3
2 2 3
```

## Sample Output 1

---

```
1
```

In this case, there are not enough apples to ship in the first box.

## Sample Input 2

---

```
2
3 3
4
1 1 3
```

```
2 1 2
1 2 3
1 1 3
```

## Sample Output 2

---

```
1
```

In this case, the amount of apples exceeds the capacity of the first box.

## Sample Input 3

---

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

## Sample Output 3

---

```
0
```

## Sample Input 4

---

```
6
28 56 99 3 125 37
10
1 1 10
1 1 14
1 3 90
1 5 10
2 3 38
2 1 5
1 3 92
1 6 18
2 5 9
2 1 4
```

## Sample Output 4

---

```
3
```



## Problem I

### 今川焼きマン - Imagawayaki Man -

#### 物語

今川焼きマンは、正義のヒーローである。顔は直径 1 メートルの今川焼き（主に小麦粉でできた生地の中にたっぷりと餡を詰めて焼いた食べ物であり、見た目は円形であり、美味しい。北海道では、単に「お焼き」と呼ばれることも多い。餡として小豆餡のほかに、カスタードクリームなどを用いることもあり、さまざまな味のバリエーションがある。そのため、いくつ食べても飽きない。小豆餡やカスタードクリームのほかに、抹茶餡やりんごジャムなども美味しい。中にタコを入れたたこ焼きのようなものもあるようだ。どの味も美味しい。下の写真は標準的な今川焼きである。左はクリーム餡、右は小豆餡が入っている。美味しそうである。実際に食べて見たところ、やはり美味しかった。ところで、今川焼きの直径は通常 10 センチメートル程度であるので、直径 1 メートルの今川焼きはかなり大きなものであることに注意されよ。）でできており、お腹の空いた者に自分の顔を分けて差し出すこともある。そのため、必要に応じて顔を取り替える必要がある。いつでも取り替えられるように、今川焼き工場でおじさんが一人でせっせと一つずつ今川焼きを焼いて日々備えている。



ところで、今川焼きは食品であるため、当然ながら賞味期限が存在する。無駄にしないためには、焼いてから時間が経ったものから順に使うことが望ましい。おじさんは下図のように、完成した直径 1 メートルの今川焼き (図中の狐色部) を幅 1 メートルのレーン (図中の灰色部) の上に隙間なく並べて保存している。



ある日、大悪党の背筋マンが工場に現れ、今川焼きマンを困らせようと、自慢の背筋力を使って巨大な今川焼きをせっせと運び、好き勝手に並び替えてしまった。背筋マンは背筋力だけでなく記憶力も良いので、どの今川焼きが元々何番目にあったのか覚えているようだ。負けを認めたくない今川焼きマンは、何とかして、それぞれの今川焼きが何番目に出来たてなのか知りたいが、見た目では分かりそうにない。そこで、背筋マンに「もちろん、僕はちゃんと覚えているよ。でも、背筋マンは本当にちゃんと覚えているのかな??」と挑発

し、さらに「本当に覚えているか僕がテストするよ！」と言って質問を繰り返すことで、何番目に出来たてなのか知ることにした。ただし、あまり質問しすぎると、探っていることを勘付かれてしまうので、ほどほどにしなければならない。あなたには、今川焼きマンを助けるため、質問を繰り返し生成し、それぞれの今川焼きが何番目に出来たてなのかを当てるプログラムを作成してほしい。

## 問題

まず、工場に並んでいる今川焼きの個数  $N$  ( $1 \leq N \leq 10,000$ ) が標準入力により 1 行で与えられる。これらの今川焼きの製造日時は互いに異なる。つまり、 $1 \leq i \leq N$  を満たす任意の整数  $i$  について、 $i$  番目に出来たての今川焼きはただひとつ存在する。

これ以降、任意の 2 整数  $a, b$  ( $1 \leq a, b \leq N$ ) において、標準出力に

```
? a b
```

と出力すると、「 $a$  番目に出来たての今川焼きの中心から  $b$  番目に出来たての今川焼きの中心までの距離は何メートルか？」という質問を行うことができる。この答えは標準入力に一行で整数としてすぐに与えられる。物語でも述べた通り、レーン上に隙間なく直径 1 メートルの今川焼きが並べられているので、左から数えて  $i$  番目の今川焼きの中心から  $j$  番目の今川焼きの中心までの距離は正確に  $|i - j|$  メートルである。この質問は 20,000 回まで繰り返し行うことができる。

それぞれの今川焼きが何番目に出来たてなのか分かったら、標準出力に最終的な答えを出力しなければならない。出力形式は、各  $i$  ( $1 \leq i \leq N$ ) において、左から  $i$  番目の今川焼きが  $x_i$  番目に出来たてのとき、

```
! x_1 x_2 x_3 ... x_N
```

または、

```
! x_N x_{N-1} x_{N-2} ... x_1
```

とすればよい。つまり、左から順に答えても、右から順に答えても構わない。この最終的な解答は、1 度しか行うことができない。この解答が正しい出力だったとき、正答とみなす。

標準出力を行う毎に、ストリームをフラッシュ (flush) する必要があることに注意されよ。主要な言語でのフラッシュ例を以下に示す。もちろん、これ以外の方法でフラッシュを行っても構わない。

C 言語:

```
#include <stdio.h>
```

```
fflush(stdout);
```

C++:

```
#include <iostream>
std::cout.flush();
```

Java:

```
System.out.flush();
```

## 入出力例

---

標準入力 標準出力

3

? 1 2

2

? 2 3

1

? 1 3

1

! 2 3 1