

# 1 A + B

1s | 512 MiB | stdin | stdout

Cho 2 số nguyên dương  $a$  và  $b$ . Hãy tính tổng  $a + b$ .

## Input

Một dòng chứa 2 số nguyên dương  $a$  và  $b$ .

## Output

In ra  $a + b$ .

## Giới hạn

$1 \leq a, b \leq 10^9$ .

## Ví dụ

stdin	stdout
1 2	3
1 2	3
1 2	3
9 7	16
1 998244353	998244354

stdin	stdout	Giải thích
1 2	3	$1 + 2 = 3$

## Giải thích

Viết giải thích nếu có

## Chấm điểm

Viết subtask nếu có

## Lưu ý

Viết lưu ý nếu có

## Minh họa

Minh họa nếu có

## Section tùy chọn

Hãy dùng cpsec từ nay về sau.

## 2 gdz

3s | 256 MiB | stdin | stdout

Đây là một bài toán tương tác.

Byteasar có một hoán vị  $(P_0, P_1, \dots, P_{n-1})$  của các số từ 0 đến  $n - 1$ .

Byteasar yêu cầu bạn đoán vị trí của số 1 trong hoán vị này. Để bạn không phải đoán một cách mù quáng, Byteasar sẽ trả lời câu hỏi "Ước số chung lớn nhất của hai số  $P_i$  và  $P_j$  là bao nhiêu?"

Để không quá đơn giản, tối đa bạn có thể hỏi  $\left\lfloor \frac{5n}{2} \right\rfloor$  câu hỏi. Ngoài ra, Byteasar liên tục đưa ra các hoán vị mới, vì vậy bạn có thể phải trả lời nhiều lần.

### Tương tác

Bạn viết một chương trình giao tiếp với Byteasar bằng cách sử dụng thư viện `gdzlib.h`, đầu tiên hãy dùng chỉ dẫn:

```
#include "gdzlib.h"
```

Thư viện này cung cấp các hàm sau:

- `int GetN()`

Trả về tham số  $n$  ( $3 \leq n \leq 5 \times 10^5$ ) hoặc  $n = -1$ . Nếu  $n \leq 3$  thì đó là độ dài hoán vị hiện tại. Nếu  $n = -1$ , nghĩa là Byteasar đã hết hoán vị.

- `int Ask(int i, int j)`

Hỏi ước số chung lớn nhất của hai số  $P_i$  và  $P_j$  ( $0 \leq i, j < n, i \neq j$ ). Trong một test, hàm này được gọi tối đa  $\left\lfloor \frac{5n}{2} \right\rfloor$  lần.

- `void Answer(int x)`

Trả lời câu hỏi của Byteasar với  $x$  là vị trí của số 1 trong hoán vị hiện tại.

Mỗi lần chạy chương trình của bạn bao gồm ít nhất một test. Chương trình phải bắt đầu bằng `GetN()` và kết thúc bằng `GetN()`. Sử dụng `Ask()` hoặc `Answer()`.

Sau bộ test cuối cùng, `GetN()` trả về  $-1$ . Khi đó, bạn phải chấm dứt chương trình.

Các test có thể có tham số  $n$  khác nhau trong một lần chạy chương trình. Tuy nhiên, số lượng hoán vị trong quá trình thực hiện chương trình sẽ không vượt quá  $5 \times 10^5$ .

Sử dụng bất kỳ hàm nào có đối số không hợp lệ sẽ bị Wrong Answer.

### 3 cho

1s | 256 MB | stdin | stdout

Sau nhiều năm, cuối cùng phù thủy Byteasar cũng tìm thấy văn bản lời cầu khẩn triệu hồi Imp Nhị Phân Vĩ Đại để trả lời câu hỏi muôn thuở của nhân loại: Liệu  $P = NP$ ?

Tuy nhiên, để hoàn thành nghi lễ, cần phải xướng tên đầy đủ của Imp, mà Byteasar chưa biết. Người ta chỉ biết rằng tên đầu tiên là một từ bao gồm đúng 100 ký tự. Đối với Imp nhị phân là tốt, đó chỉ là các ký tự 0 và 1.

Byteasar có thể thử nghi lễ nhiều lần (nhưng không quá 222 lần, sau đó cỗ máy giả kim cũ kỹ của cậu phải tạm dừng). Nếu cậu nói đúng tên Imp ít nhất một lần, nghi thức sẽ thành công và Internet toàn cầu sẽ thay đổi mãi mãi. Nhờ trực giác kỳ diệu của mình, sau mỗi lần đoán tên, Byteasar sẽ biết chính xác mình đoán đúng bao nhiêu ký tự trong tên Imp. Tuy nhiên, Imp rất thất thường, và thỉnh thoảng thay đổi tên riêng của mình. Chính xác hơn, sau mỗi lần xướng tên thất bại của Byteasar, Imp rút ngẫu nhiên (theo phân phối đều) số  $k$  từ tập  $0, 1, \dots, 99$ . Nếu ký tự thứ  $k$  được Byteasar nói đúng trong lần thử này, Imp sẽ ngay lập tức đảo ngược (0 thành 1 và 1 thành 0).

Là người học việc của một phù thủy không may mắn, bạn phải giúp Byteasar đoán tên của Imp - hãy làm điều đó nhanh chóng, nếu không Byteasar tức giận có thể biến bạn thành hòn đá.

#### Chi tiết cài đặt

Để có thể giải quyết vấn đề, cần thêm thư viện vào đầu của chương trình

```
#include "cho.h"
```

Thư viện cung cấp hàm:

```
int recytuj (const std::string& proba)
```

hoạt động như mô tả ở trên - nếu chuỗi proba giống với tên thật của Imp, chương trình của bạn thoát ngay lập tức và AC test này. Nếu không, hàm trả về một số (từ 0 đến 99) bằng số ký tự khớp với tên thật. Sau đó, nó chạy một đoạn mã sau:

```
int k = rand() % 100;
if(proba[k] == real_name[k])
    flip(real_name[k]);
```

Trong đó `real_name` là tên (bí mật) của Imp và `flip()` đảo bit  $k$  từ 0 thành 1 và ngược lại. Đối với mỗi truy vấn, đầu vào phải là một chuỗi có đúng 100 ký tự 0 hoặc 1. Ngoài ra bạn không thể gọi `recytuj()` nhiều hơn 222 lần.

Nếu bạn vi phạm bất kỳ quy tắc nào trong số này, bạn sẽ bị WA.

Tên Imp ban đầu được đặt ở đầu mỗi test.

## A + B

1s | 512 MiB | stdin | stdout

Cho 2 số nguyên dương  $a$  và  $b$ . Hãy tính tổng  $a + b$ .

### Input

Một dòng chứa 2 số nguyên dương  $a$  và  $b$ .

### Output

In ra  $a + b$ .

### Giới hạn

$1 \leq a, b \leq 10^9$ .

### Ví dụ

stdin	stdout
1 2	3

stdin	stdout	Giải thích
1 2	3	$1 + 2 = 3$
		$4 + 5 = 9$

### Giải thích

Viết giải thích nếu có

### Chấm điểm

Viết subtask nếu có

### Lưu ý

Viết lưu ý nếu có

### Minh họa

Minh họa nếu có

### Section tùy chọn

Hãy dùng cpsec từ nay về sau.