

[Đệ Quy]. Bài 1. Tính tổng 1

Tính tổng hàm $S(n) = 1 + 2 + 3 + 4 + \dots + n$ bằng đệ quy.

Gợi ý : Bài toán cơ sở : $S(1) = 1$, Công thức truy hồi : $S(n) = n + S(n - 1)$ với $n > 1$

Đầu vào

Số nguyên dương n .

Giới hạn

$0 \leq n \leq 1000$

Đầu ra

In ra $S(n)$

Ví dụ :

Input 01

Copy

```
773
```

Output 01

Copy

```
299151
```

[Đệ Quy]. Bài 2. Tính tổng 2

Tính tổng hàm $S(n) = 1^2 + 2^2 + 3^2 + 4^2 + \dots + n^2$ bằng đệ quy.

Gợi ý : Bài toán cơ sở : $S(1) = 1$, Công thức truy hồi : $S(n) = n^2 + S(n - 1)$ với $n > 1$

Đầu vào

Số nguyên dương n

Giới hạn

$0 \leq n \leq 1000$

Đầu ra

In ra kết quả của $S(n)$

Ví dụ :

Input 01

Copy

53

Output 01

Copy

51039

[Đệ Quy]. Bài 3. Tính tổng 3

Tính tổng hàm $S(n) = 1^3 + 2^3 + 3^3 + 4^3 + \dots + n^3$ bằng đệ quy

Đầu vào

Số nguyên dương n

Giới hạn

$0 \leq n \leq 10^3$

Đầu ra

In ra kết quả của $S(n)$

Ví dụ :

Input 01

Copy

435

Output 01

Copy

8992728900

[Đệ Quy]. Bài 4. Tính tổng 4

Tính tổng hàm $S(n) = -1 + 2 - 3 + 4 - 5 + 6 + \dots + (-1)^n * n$ bằng đệ quy

Gợi ý : Chia ra 2 trường hợp n chẵn và n lẻ

Đầu vào

Số nguyên dương n

Giới hạn

$1 \leq n \leq 10^3$

Đầu ra

In ra kết quả S(n)

Ví dụ :

Input 01

Copy

919

Output 01

Copy

-460

[Đệ Quy]. Bài 5. Tính giai thừa

Tính n giai thừa bằng đệ quy.

Chú ý trường hợp $n = 0$ nên bạn cần để điều kiện dừng khi $n = 0$, để khi $n = 1$ nếu input $n = 0$ sẽ không có điểm dừng đệ quy

Đầu vào

Số nguyên dương n

Giới hạn

$0 \leq n \leq 10$

Đầu ra

Kết quả của $N!$

Ví dụ :

Input 01

Copy

6

Output 01

Copy

720

[Đệ Quy]. Bài 6. Fibonacci

Dãy số **fibonacci** là dãy số thỏa mãn : $F_1=0$, $F_2=1$, $F_n=F_{n-1}+F_{n-2}$. Hãy tìm số **Fibonacci** thứ n sử dụng đệ quy.

Lưu ý : Code **Fibonacci** bằng đệ quy sẽ có độ phức tạp là $O(1.618^n)$ nên chỉ áp dụng với n nhỏ, n lớn các bạn dùng vòng lặp, mảng

Đầu vào

Số nguyên dương n .

Giới hạn

$1 \leq n \leq 15$

Đầu ra

In ra số **Fibonacci** thứ n .

Ví dụ :

Input 01

Copy

1

Output 01

Copy

0

[Đệ Quy]. Bài 7. Tổ hợp

Cho 2 số nguyên dương N và K . Hãy tính tổ hợp chập K của N . Tiện thể các bạn ôn lại một vài tính chất của tổ hợp chập K của N nhé.

Số các tổ hợp chập k của n phần tử:

$$C_n^k = \frac{n!}{k! \cdot (n - k)!}$$

Qui ước:

$$C_n^0 = 1$$

Tính chất:

$$C_n^0 = C_n^n = 1$$

$$C_n^k = C_n^{n-k}$$

$$C_n^k = C_{n-1}^{k-1} + C_{n-1}^k$$

$$C_n^k = \frac{n - k + 1}{k} C_n^{k-1}$$

Đầu vào

2 số nguyên dương N và K .

Giới hạn

$0 \leq k \leq n \leq 10$.

Đầu ra

Kết quả của tổ hợp chập K của N .

Ví dụ :

Input 01

Copy

10 2

Output 01

Copy

45

[Đệ Quy]. Bài 8. UCLN & BCNN

Cho 2 số **a** và **b**, hãy tính ước chung lớn nhất và bội chung nhỏ nhất của 2 số **a** và **b**. Trong đó hàm **UCLN** sử dụng đệ quy để tính.

Gợi ý : $BCNN(a, b) = a / UCLN * b$ thay vì $a * b / UCLN$ để tránh việc lấy $a * b$ bị tràn long long

Đầu vào

2 số nguyên dương **a** và **b**.

Giới hạn

$1 \leq a, b \leq 10^{12}$

Đầu ra

In ra **UCLN** và **BCNN** của 2 số. Dữ liệu đảm bảo **UCLN** của 2 số nằm trong khoảng số nguyên 64 bit.

Ví dụ :

Input 01

Copy

10 20

Output 01

Copy

10 20

[Độ Quy]. Bài 9. Lũy thừa nhị phân

Cho 2 số nguyên không âm **a** và **b**. Hãy tính **$a^b \%(10^9+7)$** . Kiến thức bạn cần sử dụng là Binary Exponentiation.

Fast Modular Exponentiation

$$(a * b) \bmod n = ((a \bmod n) * (b \bmod n)) \bmod n$$

$$X^Y \bmod N = \begin{cases} (X^{\lfloor \frac{Y}{2} \rfloor} \bmod N)^2 \bmod N & \text{if } Y \text{ is even} \\ ((X^{\lfloor \frac{Y}{2} \rfloor} \bmod N)^2 \bmod N) * X \bmod N & \text{if } Y \text{ is odd} \end{cases}$$

We can save the value of $(X^{\lfloor \frac{Y}{2} \rfloor} \bmod N)$ to not compute it twice
 $O(\log(n))$

Đầu vào

2 số nguyên dương **a, b**

Giới hạn

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

Đầu ra

In ra kết quả của bài toán.

Ví dụ :

Input 01

Copy

2 10

Output 01

Copy

1024

[Độ Quy]. Bài 10. Tính tổng 5

Tính tổng $S(n) = 1/1 + 1/2 + 1/3 + \dots + 1/n$ bằng đệ quy.

Đầu vào

Số nguyên dương n .

Giới hạn

$1 \leq n \leq 1000$

Đầu ra

In ra $S(n)$ lấy 3 số sau dấu phẩy.

Ví dụ :

Input 01

Copy

18

Output 01

Copy

```
3.495
```

[Đề Quy]. Bài 11. Thập phân sang nhị phân

Cho một số nguyên không âm **N**, hãy in ra dạng biểu diễn nhị phân của số **N**.

Đầu vào

Số nguyên không âm **N**

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

Biểu diễn nhị phân của số nguyên **N**.

Ví dụ :

Input 01

Copy

```
0
```

Output 01

Copy

```
0
```

Input 02

Copy

7

Output 02

Copy

111

[Đệ Quy]. Bài 12. Thập phân sang hệ thập lục phân

Cho một số nguyên không âm **N**, hãy in ra dạng biểu diễn của **N** dưới hệ 16.

Gợi ý : Để điều kiện dừng là khi $N < 16$ sẽ dễ xử lý trường hợp khi $N = 0$ hơn

Đầu vào

Số nguyên không âm **N**

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

Biểu diễn hệ 16 của số nguyên **N**

Ví dụ :

Input 01

Copy

995

Output 01

Copy

3E3

Input 02

Copy

0

Output 02

Copy

0

[Đệ Quy]. Bài 13. Tìm tổng chữ số của N

Cho một số nguyên không âm **N**, hãy tính tổng chữ số của **N** sử dụng kỹ thuật đệ quy.

Đầu vào

Số nguyên không âm **N**

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

Tổng các chữ số của **N**

Ví dụ :

Input 01

Copy

123456789

Output 01

Copy

[Đệ Quy]. Bài 14. Đếm số chữ số của N

Cho một số nguyên không âm **N**, hãy đếm số lượng chữ số của **N**.

Đầu vào

Số nguyên không âm **N**

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

Số lượng chữ số của **N**.

Ví dụ :

Input 01

Copy

```
123452
```

Output 01

Copy

```
6
```

[Đệ Quy]. Bài 15. Tìm chữ số đầu tiên của số nguyên

Cho một số nguyên không âm **N**, hãy in ra chữ số đầu tiên của **N**.

Đầu vào

Số nguyên không âm **N**

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

In ra chữ số đầu tiên của **N**.

Ví dụ :

Input 01

Copy

56721

Output 01

Copy

5

[Đệ Quy]. Bài 16. Chữ số lớn nhất, nhỏ nhất

Cho một số nguyên không âm **N**, hãy in ra chữ số lớn nhất và chữ số nhỏ nhất của **N**.

Gợi ý : Các bạn viết 2 hàm tìm chữ số nhỏ nhất và lớn nhất.

Đầu vào

Số nguyên dương **N**

Giới hạn

$10 \leq n \leq 10^{18}$

Đầu ra

Chữ số lớn nhất và nhỏ nhất của **N**.

Ví dụ :

Input 01

Copy

```
1256782
```

Output 01

Copy

```
8 1
```

[Đệ Quy]. Bài 17. In ra số nguyên

Cho một số nguyên không âm **N**, hãy in ra **N** theo thứ tự các chữ số từ trái qua phải và từ phải qua trái. Chú ý phải sử dụng hàm đệ quy cho cả 2 yêu cầu.

Đầu vào

Số nguyên không âm **N**.

Giới hạn

$$0 \leq n \leq 10^{18}$$

Đầu ra

Dòng đầu tiên in ra các chữ số của **n** theo thứ tự từ trái qua phải.

Dòng thứ hai in ra các chữ số của **n** theo thứ tự từ phải qua trái. Các chữ số được viết cách nhau một dấu cách.

Ví dụ :

Input 01

Copy

```
21218
```

Output 01

Copy

```
2 1 2 1 8
8 1 2 1 2
```

[Đệ Quy]. Bài 18. Tính tổng chữ số chẵn lẻ

Cho số nguyên không âm **N**, hãy tính tổng các chữ số chẵn, tổng các chữ số lẻ của **N**.

Bạn sẽ viết 2 hàm, 1 hàm tính tổng chẵn, 1 hàm tính tổng lẻ.

Đầu vào

Số nguyên không âm **N**.

Giới hạn

$0 \leq n \leq 10^{18}$

Đầu ra

Dòng đầu tiên in ra tổng các chữ số chẵn.

Dòng thứ 2 in ra tổng các chữ số lẻ.

Ví dụ :

Input 01

Copy

```
123456
```

Output 01

Copy

```
12
9
```


[Đệ Quy]. Bài 19. Kiểm tra chữ số chẵn

Cho số nguyên không âm **N**, hãy kiểm tra xem tất cả các chữ số của **N** có phải đều là số chẵn hay không?

Đầu vào

Số nguyên không âm **N**.

Giới hạn

$$0 \leq n \leq 10^{18}$$

Đầu ra

In ra **YES** nếu **n** toàn chữ số chẵn, ngược lại in ra **NO**.

Ví dụ :

Input 01

Copy

```
2280820
```

Output 01

Copy

```
YES
```

[Đệ Quy]. Bài 20. Đếm số thao tác

Cho số nguyên dương **N**. Bạn được thực hiện 3 thao tác sau đây:

1. Lấy **N** chia cho 2 nếu **N** chia hết cho 2.
2. Lấy **N** chia hết cho 3 nếu **N** chia hết cho 3.
3. Giảm **N** đi 1 đơn vị. Hãy đếm số thao tác ít nhất để biến đổi **N** về 1.

Gợi ý : Gọi **F(n)** là hàm đếm số thao tác biến đổi **n** về 1, khi đó $F(n) = \min(F(n/2), F(n/3), F(n-1))$ chứ không được return về 1 trong 3 hàm kia. Bạn cần gọi 3

hàm đệ quy tương ứng với 3 thao tác, sau đó trả về số nhỏ nhất trong 3 hàm đó

Đầu vào

Số nguyên dương **N**.

Giới hạn

$1 \leq n \leq 50$

Đầu ra

In ra số thao tác tối thiểu cần thực hiện.

Ví dụ :

Input 01

Copy

38

Output 01

Copy

5

Giải thích :

1. $N = N / 2 = 19$ 2. $N = N - 1 = 18$ 3. $N = N / 3 = 6$ 4. $N = N / 2 = 3$ 5. $N = N / 3 = 1$

Input 02

Copy

10

Output 02

Copy

3

Giải thích :

1. $N = N - 1 = 9$ 2. $N = N / 3 = 3$ 3. $N = N / 3 = 1$

[Đệ Quy]. Bài 21. Kiểm tra mảng đối xứng

Cho một mảng số nguyên **A** gồm **N** phần tử, hãy kiểm tra xem mảng có đối xứng hay không bằng cách sử dụng hàm đệ quy.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Giới hạn

$1 \leq n \leq 1000$

$1 \leq A[i] \leq 10^6$

Đầu ra

In ra **YES** nếu mảng **A[]** là mảng đối xứng, ngược lại in ra **NO**.

Ví dụ :

Input 01

Copy

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

Output 01

Copy

YES

[Đệ Quy]. Bài 22. In mảng

Cho một mảng số nguyên **A** gồm **N** phần tử, hãy in ra mảng theo thứ tự từ trái qua phải và từ phải qua trái bằng đệ quy. Bạn sẽ viết 2 hàm tương ứng với 2 yêu cầu.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Giới hạn

$$1 \leq n \leq 1000$$
$$1 \leq A[i] \leq 10^6$$

Đầu ra

Dòng đầu tiên in ra các phần tử trong mảng theo chiều từ trái qua phải.

Dòng thứ 2 in ra các phần tử trong mảng theo chiều từ phải qua trái.

Ví dụ :

Input 01

Copy

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

Output 01

Copy

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

[Đệ Quy]. Bài 23. Kiểm tra mảng toàn chẵn

Cho một mảng số nguyên **A** gồm **N** phần tử, hãy kiểm tra xem mảng có phải toàn số chẵn hay không?

Đầu vào

Dòng đầu tiên là số nguyên dương **N**

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Giới hạn

$$1 \leq n \leq 1000$$

$$1 \leq A[i] \leq 10^6$$

Đầu ra

In ra **YES** nếu mảng toàn số chẵn, ngược lại in ra **NO**.

Ví dụ :

Input 01

Copy

```
4
2 8 10 20012
```

Output 01

Copy

```
YES
```

[Đệ Quy]. Bài 24. Kiểm tra mảng tăng dần

Cho một mảng số nguyên **A** gồm **N** phần tử, hãy kiểm tra xem mảng có phải là mảng tăng dần hay không, mảng tăng dần được định nghĩa là mảng mà phần tử hiện tại luôn lớn hơn các phần tử đứng trước nó.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Giới hạn

$$1 \leq n \leq 1000$$

$$1 \leq A[i] \leq 10^6$$

Đầu ra

In ra **YES** nếu mảng toàn là mảng tăng dần, ngược lại in ra **NO**.

Ví dụ :

Input 01

Copy

```
5
1 1 8 9 20
```

Output 01

Copy

```
NO
```

[Độ Quy]. Bài 25. Binary search

Cho một mảng số nguyên **A** gồm **N** phần tử đã được sắp xếp **giảm dần**, hãy viết hàm tìm kiếm nhị phân bằng đệ quy để kiểm tra xem phần tử **X** có nằm trong mảng hay không.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**.

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Dòng thứ 3 là số nguyên dương **X**.

Giới hạn

$1 \leq n \leq 1000$

$1 \leq X, A[i] \leq 10^6$

Đầu ra

In ra 1 nếu **X** xuất hiện trong mảng, ngược lại in ra 0.

Ví dụ :

Input 01

Copy

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

Output 01

Copy

```
1
```

[Đệ Quy]. Bài 26. Lower bound

Cho một mảng số nguyên **A** gồm **N** phần tử đã được sắp xếp tăng dần, hãy viết hàm lower bound bằng đệ quy giúp tìm vị trí đầu tiên của phần tử lớn hơn hoặc bằng phần tử **X** cho trước hoặc xác định rằng không có phần tử nào trong mảng lớn hơn hoặc bằng **X**. Tương tự như thuật toán Binary search, độ phức tạp của hàm lower_bound này cũng là $O(\log N)$.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**.

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Dòng thứ 3 là số nguyên dương **X**.

Giới hạn

$$1 \leq n \leq 1000$$

$$1 \leq X, A[i] \leq 10^6$$

Đầu ra

In ra vị trí đầu tiên của phần tử lớn hơn hoặc bằng **X** trong mảng, nếu trong mảng không có phần tử nào lớn hơn hoặc bằng **X** thì in ra **N**.

Ví dụ :

Input 01

Copy

```
6
1 2 3 3 3 4
3
```

Output 01

Copy

```
2
```

[Đệ Quy]. Bài 27. Upper bound

Cho một mảng số nguyên **A** gồm **N** phần tử đã được sắp xếp giảm dần, hãy viết hàm lower bound bằng đệ quy giúp tìm vị trí cuối cùng của phần tử lớn hơn phần tử **X** cho trước hoặc xác định rằng không có phần tử nào trong mảng lớn hơn **X**. Tương tự như thuật toán Binary search, độ phức tạp của hàm upper_bound này cũng là $O(\log N)$.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**.

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Dòng thứ 3 là số nguyên dương **X**.

Giới hạn

$$1 \leq n \leq 1000$$

$$1 \leq X, A[i] \leq 10^6$$

Đầu ra

In ra vị trí cuối cùng của phần tử lớn hơn **X** trong mảng, nếu trong mảng không có phần tử nào lớn hơn **X** thì in ra **N**.

Ví dụ :

Input 01

Copy

```
6
6 5 5 4 3 2
4
```

Output 01

Copy

```
2
```

[Đệ Quy]. Bài 28. Số lần xuất hiện

Cho một mảng số nguyên **A** gồm **N** phần tử đã được sắp xếp tăng dần, hãy đếm số lần xuất hiện của phần tử **X** trong mảng bằng cách viết hàm tìm vị trí xuất hiện đầu tiên của **X** trong mảng, hàm tìm vị trí xuất hiện cuối cùng của **x** trong mảng, sau đó lấy vị trí cuối trừ vị trí đầu cộng thêm 1 sẽ ra số lần xuất hiện của **X** trong mảng. Như vậy độ phức tạp của thuật toán sẽ là $O(\log N)$.

Bạn cần viết 2 hàm tìm vị trí đầu tiên và cuối cùng

Đầu vào

Dòng đầu tiên là số nguyên dương **N**.

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Dòng thứ 3 là số nguyên dương **X**.

Giới hạn

$1 \leq n \leq 1000$

$1 \leq X, A[i] \leq 10^6$

Đầu ra

In ra số lần xuất hiện của **X** trong mảng.

Ví dụ :

Input 01

Copy

```
6
1 2 2 3 3 4
2
```

Output 01

Copy

```
2
```

[Đệ Quy]. Bài 29. Tháp Hà Nội

Bài toán Tháp Hà Nội bao gồm ba cọc (1, 2 và 3) và n đĩa tròn có kích thước khác nhau. Ban đầu, cọc 1 có tất cả các đĩa, theo thứ tự kích thước tăng dần từ trên xuống dưới. Yêu cầu là để di chuyển tất cả các đĩa sang cọc thứ 3 bằng cách sử dụng cọc ở giữa. Trên mỗi lần di chuyển, bạn có thể di chuyển đĩa trên cùng từ cọc này sang cọc khác. Ngoài ra, không được phép đặt một đĩa lớn hơn trên một đĩa nhỏ hơn trong quá trình di chuyển các đĩa. Nhiệm vụ của bạn là tìm ra giải pháp giảm thiểu số lần di chuyển.

Đầu vào

Số nguyên dương n là số lượng đĩa ban đầu ở cọc 1.

Giới hạn

$$1 \leq n \leq 10$$

Đầu ra

Dòng đầu tiên in số lượng lần di chuyển các đĩa tối thiểu. Các dòng tiếp theo mỗi dòng ghi ra 2 cọc tương ứng với thứ tự chuyển đĩa ở bước đó.

Ví dụ :

Input 01

Copy

```
2
```

Output 01

Copy

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

[Đệ Quy]. Bài 30. Merge sort

Cho một mảng số nguyên **A** gồm **N** phần tử, hãy sắp xếp mảng tăng dần bằng thuật toán sắp xếp trộn.

Đầu vào

Dòng đầu tiên là số nguyên dương **N**

Dòng thứ 2 là **N** số nguyên trong mảng được viết cách nhau một dấu cách.

Giới hạn

$$1 \leq n \leq 1000$$

$$1 \leq A[i] \leq 10^6$$

Đầu ra

In ra mảng được sắp xếp theo thứ tự tăng dần.

Ví dụ :

Input 01

Copy

```
5
6 7 8 1 2
```

Output 01

Copy

```
1 2 6 7 8
```