

## ✓ C++ Buổi 04\_Bài 01.Luyện tập viết hàm.

Cho số nguyên n không âm. Viết hàm xử lý các yêu cầu sau:

1. Kiểm tra n là số nguyên tố, nếu đúng in 1, sai in 0.
2. In tổng chữ số của n.
3. In tổng chữ số chẵn của n.
4. In tổng chữ số của n là số nguyên tố.
5. In số lật ngược của n. Ví dụ 123 in 321.
6. In số lượng ước riêng biệt của n là số nguyên tố (làm tương tự như phân tích thừa số nguyên tố).
7. In ước nguyên tố lớn nhất của n (làm tương tự như phân tích thừa số nguyên tố).
8. Kiểm tra nếu n tồn tại ít nhất 1 số 6, nếu đúng in 1, sai in 0.
9. Kiểm tra nếu tổng chữ số của n chia hết cho 8, nếu đúng in 1, sai in 0.
10. Tính tổng giai thừa các chữ số của n và in ra. Ví dụ n = 123, tổng =  $1! + 2! + 3!$
11. Kiểm tra n có phải chỉ được tạo bởi 1 số hay không? Ví dụ 222, 333, 99999. Đúng in ra 1, sai in ra 0.
12. Kiểm tra n có phải có chữ số tận cùng là lớn nhất hay không, tức là không có chữ số nào của n lớn hơn chữ số hàng đơn vị của nó. nếu đúng in 1, sai in 0.
13. In tổng lũy thừa chữ số của n với cơ số là số chữ số. ví dụ 123 thì tính  $1^3 + 2^3 + 3^3$ .

### Dữ liệu vào:

Số duy nhất  $n$ . ( $2 \leq n \leq 10^5$ ).

### Dữ liệu ra:

In ra 13 dòng tương ứng với các yêu cầu ở trên.

### Ví dụ:

#### Dữ liệu vào:

37

#### Dữ liệu ra:

1  
10  
0  
10  
73  
1  
37  
0  
0  
5046  
0  
1  
58

## ✓ C++ Buổi 04\_Bài 02.Kiểm tra số nguyên tố.

Kiểm tra một số nguyên không âm N có phải là số nguyên tố hay không?

### Dữ liệu vào:

Dòng duy nhất chứa số nguyên dương N. ( $0 \leq N \leq 10^9$ )

### Dữ liệu ra:

In ra YES nếu n là số nguyên tố, ngược lại in NO.

### Ví dụ:

#### Dữ liệu vào:

7

#### Dữ liệu ra:

YES

## ✓ C++ Buổi 04\_Bài 03.Kiểm tra số thuần số nguyên tố.

Một số được coi là thuần nguyên tố nếu nó là số nguyên tố, tất cả các chữ số là nguyên tố và tổng chữ số của nó cũng là một số nguyên tố. Bài toán đặt ra là đếm xem trong một đoạn giữa hai số nguyên cho trước có bao nhiêu số thuần nguyên tố.

### Dữ liệu vào:

Nhập 2 số nguyên dương. ( $1 \leq a, b \leq 10^6$ ).

### Dữ liệu ra:

In ra kết quả của bài toán.

### Ví dụ:

#### Dữ liệu vào:

```
1 10
```

[Copy](#)

#### Dữ liệu ra:

```
4
```

[Copy](#)

## ✓ C++ Buổi 04\_Bài 04.Kiểm tra số chính phương.

Kiểm tra một số nguyên có phải là số chính phương hay không?

### Dữ liệu vào:

Một số nguyên dương  $N$ . ( $1 \leq N \leq 10^{18}$ ).

### Dữ liệu ra:

In ra YES nếu  $N$  là số chính phương, ngược lại in NO

### Ví dụ:

#### Dữ liệu vào:

16

#### Dữ liệu ra:

YES

## ✓ C++ Buổi 04\_Bài 05.Số Chính Phương trong đoạn.

In ra các số chính phương trong đoạn từ a tới b.

### Dữ liệu vào:

2 số nguyên dương a, b. ( $1 \leq a \leq b \leq 10^{10}$ ).

### Dữ liệu ra:

In ra các số chính phương trong đoạn giữa 2 số a, b trên một dòng. Các số cách nhau một khoảng trắng.

### Ví dụ:

#### Dữ liệu vào:

2 20

#### Dữ liệu ra:

4 9 16

## ✔ C++ Buổi 04\_Bài 06.Ước chung lớn nhất, bội chung nhỏ nhất.

Tìm ước chung lớn nhất và bội chung nhỏ nhất của 2 số nguyên.

### Dữ liệu vào:

2 số nguyên a, b. ( $1 \leq a, b \leq 10^{12}$ ).

### Dữ liệu ra:

In ra ước chung lớn nhất và bội chung nhỏ nhất.

### Ví dụ:

#### Dữ liệu vào:

20 50

#### Dữ liệu ra:

10 100

## ✓ C++ Buổi 04\_Bài 07.Phân tích thừa số nguyên tố.

Hãy phân tích một số nguyên dương N thành thừa số nguyên tố.

### Dữ liệu vào:

Số nguyên dương N. ( $1 \leq N \leq 10^{16}$ ).

### Dữ liệu ra:

Phân tích thừa số nguyên tố của N, xem ví dụ để rõ hơn format.

### Ví dụ:

#### Dữ liệu vào:

30

#### Dữ liệu ra:

$2^1 * 3^1 * 5^1$



## ✓ C++ Buổi 04\_Bài 08.Ước số nguyên tố nhỏ nhất.

Cho số tự nhiên  $N$ . Nhiệm vụ của bạn là in ra ước số nguyên tố nhỏ nhất của các số từ 1 đến  $N$ . Ước số nguyên tố nhỏ nhất của 1 là 1. Ước số nguyên tố nhỏ nhất của các số chẵn là 2. Ước số nguyên tố nhỏ nhất của các số nguyên tố là chính nó.

### Dữ liệu vào:

Một số  $N$  được ghi trên một dòng. ( $1 \leq N \leq 100000$ ).

### Dữ liệu ra:

Đưa ra kết quả theo từng dòng.

### Ví dụ:

#### Dữ liệu vào:

```
7
```

[Copy](#)

#### Dữ liệu ra:

```
1
2
3
2
5
2
7
```

[Copy](#)

## ✓ C++ Buổi 04\_Bài 09.Ước số nguyên tố lớn nhất.

Tìm ước số nguyên tố lớn nhất của một số nguyên dương.

### Dữ liệu vào:

Dòng đầu tiên là số lượng test case T; T dòng tiếp theo mỗi dòng là một số nguyên dương N. ( $1 \leq T \leq 500$ ;  $2 \leq N \leq 10000000$ ).

### Dữ liệu ra:

Ước số nguyên tố lớn nhất của n in ra mỗi test case trên 1 dòng

### Ví dụ:

#### Dữ liệu vào:

```
2
13
10
```

#### Dữ liệu ra:

```
13
5
```

## ✓ C++ Buổi 04\_Bài 10. Thừa số nguyên tố thứ K.

Đưa ra số nguyên tố thứ k trong phân tích thừa số nguyên tố của một số nguyên dương n. Ví dụ n=28, k=3 ta có kết quả là 7 vì  $28=2 \times 2 \times 7$ .

### Dữ liệu vào:

2 số n, k ( $1 \leq n, k \leq 10^9$ ).

### Dữ liệu ra:

In ra thừa số nguyên tố thứ k của n, nếu n không có thừa số nguyên tố thứ k thì in ra -1.

### Ví dụ:

#### Dữ liệu vào:

28 3

#### Dữ liệu ra:

7

## ✓ C++ Buổi 04\_Bài 11.Goldbach conjecture.

Theo Goldbach conjecture, một số nguyên dương chẵn  $\geq 4$  đều có thể biểu diễn dưới dạng tổng của 2 số nguyên tố. Cho số nguyên dương chẵn  $N \geq 4$ . Hãy liệt kê các cặp số nguyên tố  $p, q$  có tổng đúng bằng  $N$ . Ví dụ  $N = 4$  ta có 1 cặp số nguyên tố là  $2 + 2 = 4$ .

### Dữ liệu vào:

Đầu vào là một số chẵn  $N$ . ( $4 \leq N \leq 1000000$ ).

### Dữ liệu ra:

Đưa ra kết quả mỗi test theo từng dòng, mỗi dòng là một cặp số thỏa mãn. Chú ý mỗi cặp chỉ liệt kê một lần, không xét đến thứ tự.

### Ví dụ:

#### Dữ liệu vào:

10

Copy

#### Dữ liệu ra:

3 7

5 5

Copy

## ✓ C++ Buổi 04\_Bài 12.Cặp số nguyên tố cùng nhau.

Hãy liệt kê các cặp số nguyên tố cùng nhau và có giá trị khác nhau trong đoạn  $[a,b]$  theo thứ tự từ nhỏ đến lớn.

### Dữ liệu vào:

Chỉ có một dòng ghi hai số  $a, b$ . ( $2 \leq a \leq b \leq 1000$ ).

### Dữ liệu ra:

Các cặp số  $ij$  thỏa mãn được viết lần lượt trên từng dòng theo định dạng  $(i,j)$ , theo thứ tự từ điển.

**Dữ liệu vào:**

2 40

**Dữ liệu ra:**

(2,3)  
(2,5)  
(2,7)  
(2,9)  
(2,11)  
(2,13)  
(2,15)  
(2,17)  
(2,19)  
(2,21)  
(2,23)  
(2,25)  
(2,27)  
(2,29)  
(2,31)  
(2,33)  
(2,35)  
(2,37)  
(2,39)  
(3,4)  
(3,5)  
(3,7)  
(3,8)  
(3,10)  
(3,11)  
(3,13)

(3,14)  
(3,16)  
(3,17)  
(3,19)  
(3,20)  
(3,22)  
(3,23)  
(3,25)  
(3,26)  
(3,28)  
(3,29)  
(3,31)  
(3,32)  
(3,34)  
(3,35)  
(3,37)  
(3,38)  
(3,40)  
(4,5)  
(4,7)  
(4,9)  
(4,11)  
(4,13)  
(4,15)  
(4,17)  
(4,19)  
(4,21)  
(4,23)  
(4,25)  
(4,27)  
(4,29)  
(4,31)

(4,33)

(4,35)

(4,37)

(4,39)

(5,6)

(5,7)

(5,8)

(5,9)

(5,11)

(5,12)

(5,13)

(5,14)

(5,16)

(5,17)

(5,18)

(5,19)

(5,21)

(5,22)

(5,23)

(5,24)

(5,26)

(5,27)

(5,28)

(5,29)

(5,31)

(5,32)

(5,33)

(5,34)

(5,36)

(5,37)

(5,38)



(5,39)  
(6,7)  
(6,11)  
(6,13)  
(6,17)  
(6,19)  
(6,23)  
(6,25)  
(6,29)  
(6,31)  
(6,35)  
(6,37)  
(7,8)  
(7,9)  
(7,10)  
(7,11)  
(7,12)  
(7,13)  
(7,15)  
(7,16)  
(7,17)  
(7,18)  
(7,19)  
(7,20)  
(7,22)  
(7,23)  
(7,24)  
(7,25)  
(7,26)  
(7,27)  
(7,29)  
(7,30)

(7,31)
(7,32)
(7,33)
(7,34)
(7,36)
(7,37)
(7,38)
(7,39)
(7,40)
(8,9)
(8,11)
(8,13)
(8,15)
(8,17)
(8,19)
(8,21)
(8,23)
(8,25)
(8,27)
(8,29)
(8,31)
(8,33)
(8,35)
(8,37)
(8,39)
(9,10)
(9,11)
(9,13)
(9,14)
(9,16)
(9,17)
(9,19)

(9,20)  
(9,22)  
(9,23)  
(9,25)  
(9,26)  
(9,28)  
(9,29)  
(9,31)  
(9,32)  
(9,34)  
(9,35)  
(9,37)  
(9,38)  
(9,40)  
(10,11)  
(10,13)  
(10,17)  
(10,19)  
(10,21)  
(10,23)  
(10,27)  
(10,29)  
(10,31)  
(10,33)  
(10,37)  
(10,39)  
(11,12)  
(11,13)  
(11,14)  
(11,15)  
(11,16)  
(11,17)

(11,18)
(11,19)
(11,20)
(11,21)
(11,23)
(11,24)
(11,25)
(11,26)
(11,27)
(11,28)
(11,29)
(11,30)
(11,31)
(11,32)
(11,34)
(11,35)
(11,36)
(11,37)
(11,38)
(11,39)
(11,40)
(12,13)
(12,17)
(12,19)
(12,23)
(12,25)
(12,29)
(12,31)
(12,35)
(12,37)
(13,14)

	(13,15)
	(13,16)
	(13,17)
	(13,18)
	(13,19)
	(13,20)
	(13,21)
	(13,22)
	(13,23)
	(13,24)
	(13,25)
	(13,27)
	(13,28)
	(13,29)
	(13,30)
	(13,31)
	(13,32)
	(13,33)
	(13,34)
	(13,35)
	(13,36)
	(13,37)
	(13,38)
	(13,40)
	(14,15)
	(14,17)
	(14,19)
	(14,23)
	(14,25)
	(14,27)
	(14,29)
	(14,31)

(14,33)  
(14,37)  
(14,39)  
(15,16)  
(15,17)  
(15,19)  
(15,22)  
(15,23)  
(15,26)  
(15,28)  
(15,29)  
(15,31)  
(15,32)  
(15,34)  
(15,37)  
(15,38)  
(16,17)  
(16,19)  
(16,21)  
(16,23)  
(16,25)  
(16,27)  
(16,29)  
(16,31)  
(16,33)  
(16,35)  
(16,37)  
(16,39)  
(17,18)  
(17,19)  
(17,20)  
(17,21)

(17,22)  
(17,23)  
(17,24)  
(17,25)  
(17,26)  
(17,27)  
(17,28)  
(17,29)  
(17,30)  
(17,31)  
(17,32)  
(17,33)  
(17,35)  
(17,36)  
(17,37)  
(17,38)  
(17,39)  
(17,40)  
(18,19)  
(18,23)  
(18,25)  
(18,29)  
(18,31)  
(18,35)  
(18,37)  
(19,20)  
(19,21)  
(19,22)  
(19,23)  
(19,24)  
(19,25)  
(19,26)

(19,27)  
(19,28)  
(19,29)  
(19,30)  
(19,31)  
(19,32)  
(19,33)  
(19,34)  
(19,35)  
(19,36)  
(19,37)  
(19,39)  
(19,40)  
(20,21)  
(20,23)  
(20,27)  
(20,29)  
(20,31)  
(20,33)  
(20,37)  
(20,39)  
(21,22)  
(21,23)  
(21,25)  
(21,26)  
(21,29)  
(21,31)  
(21,32)  
(21,34)  
(21,37)  
(21,38)



(21,40)  
(22,23)  
(22,25)  
(22,27)  
(22,29)  
(22,31)  
(22,35)  
(22,37)  
(22,39)  
(23,24)  
(23,25)  
(23,26)  
(23,27)  
(23,28)  
(23,29)  
(23,30)  
(23,31)  
(23,32)  
(23,33)  
(23,34)  
(23,35)  
(23,36)  
(23,37)  
(23,38)  
(23,39)  
(23,40)  
(24,25)  
(24,29)  
(24,31)  
(24,35)  
(24,37)  
(25,26)

(25,27)  
(25,28)  
(25,29)  
(25,31)  
(25,32)  
(25,33)  
(25,34)  
(25,36)  
(25,37)  
(25,38)  
(25,39)  
(26,27)  
(26,29)  
(26,31)  
(26,33)  
(26,35)  
(26,37)  
(27,28)  
(27,29)  
(27,31)  
(27,32)  
(27,34)  
(27,35)  
(27,37)  
(27,38)  
(27,40)  
(28,29)  
(28,31)  
(28,33)  
(28,37)  
(28,39)  
(29,30)

(29,31)  
(29,32)  
(29,33)  
(29,34)  
(29,35)  
(29,36)  
(29,37)  
(29,38)  
(29,39)  
(29,40)  
(30,31)  
(30,37)  
(31,32)  
(31,33)  
(31,34)  
(31,35)  
(31,36)  
(31,37)  
(31,38)  
(31,39)  
(31,40)  
(32,33)  
(32,35)  
(32,37)  
(32,39)  
(33,34)  
(33,35)  
(33,37)  
(33,38)  
(33,40)  
(34,35)

(34, 37)

(34, 39)

(35, 36)

(35, 37)

(35, 38)

(35, 39)

(36, 37)

(37, 38)

(37, 39)

(37, 40)

(38, 39)

(39, 40)

## ✓ C++ Buổi 04\_Bài 13.Số thuận nghịch có 3 ước nguyên tố.

Một số được coi là đẹp nếu nó là số thuận nghịch và có ít nhất 3 ước số nguyên tố khác nhau. Viết chương trình in ra các số đẹp như vậy trong một đoạn giữa hai số nguyên cho trước.

### Dữ liệu vào:

2 số  $a, b (1 \leq a \leq b \leq 10^7)$ .

### Dữ liệu ra:

In ra các số đẹp trong đoạn  $a, b$ . Trong trường hợp không tồn tại số đẹp nào thì in ra -1.

### Ví dụ:

#### Dữ liệu vào:

1 1000

#### Dữ liệu ra:

66 222 252 282 414 434 444 474 494 525 555 585 595 606 616 636 646 666 696 777 828 858 868 888 969

## ✓ C++ Buổi 04\_Bài 14.Số thuận nghịch, lọc phát.

Một số được coi là số đẹp nếu nó là số thuận nghịch, có chứa ít nhất một chữ số 6, và tổng các chữ số của nó có chữ số cuối cùng là 8. Viết chương trình liệt kê các số đẹp trong đoạn giữa 2 số nguyên cho trước, các số cách nhau một dấu cách.Nếu trong đoạn không có số đẹp thì in ra -1.

### Dữ liệu vào:

2 số nguyên a, b.( $1 \leq a \leq b \leq 10^6$ ).

### Dữ liệu ra:

Liệt kê các số đẹp trong đoạn, các số viết cách nhau một khoảng trống.

### Ví dụ:

#### Dữ liệu vào:

100 300

Copy

#### Dữ liệu ra:

161

Copy

## ✓ C++ Buổi 04\_Bài 15.Số Sphenic.

Số nguyên dương  $N$  được gọi là số Sphenic nếu  $N$  được phân tích duy nhất dưới dạng tích của ba thừa số nguyên tố khác nhau. Ví dụ  $N=30$  là số Sphenic vì  $30 = 2 \times 3 \times 5$ ;  $N = 60$  không phải số Sphenic vì  $60 = 2 \times 2 \times 3 \times 5$ . Cho số tự nhiên  $N$ , nhiệm vụ của bạn là kiểm tra xem  $N$  có phải số Sphenic hay không?

### Dữ liệu vào:

Một số nguyên dương  $N$ . ( $1 \leq N \leq 10^{18}$ ).

### Dữ liệu ra:

Đưa ra YES hoặc NO tương ứng với  $N$  là số Sphenic hoặc không.

### Ví dụ:

#### Dữ liệu vào:

30

#### Dữ liệu ra:

YES

## ✓ C++ Buổi 04\_Bài 16.Số Smith.

Cho số tự nhiên  $N$ . Nhiệm vụ của bạn là hãy kiểm tra  $N$  có phải là số Smith hay không. Một số được gọi là số Smith nếu  $N$  không phải là số nguyên tố và có tổng các chữ số của  $N$  bằng tổng các chữ số của các thừa số nguyên tố trong phân tích của  $N$ . Ví dụ  $N = 666$  có các thừa số nguyên tố là 2, 3, 3, 37 có tổng các chữ số là 18.

### Dữ liệu vào:

Số nguyên dương  $N$ . ( $1 \leq N \leq 10^8$ ).

### Dữ liệu ra:

In ra YES nếu  $N$  là số Smith, ngược lại in ra NO.

### Ví dụ:

#### Dữ liệu vào:

22

Copy

#### Dữ liệu ra:

YES

Copy



## ✔ C++ Buổi 04\_Bài 17.Số Hoàn Hảo.

Số hoàn hảo là số có tổng các ước thực sự (Không tính chính nó) bằng chính số đó. Cho một số nguyên dương  $n$ , kiểm tra xem  $n$  có phải là số hoàn hảo hay không.

### Dữ liệu vào:

Số nguyên dương  $N(1 \leq N \leq 9 \cdot 10^{18})$ .

### Dữ liệu ra:

In YES nếu  $N$  là số hoàn hảo, ngược lại in NO.

### Ví dụ:

#### Dữ liệu vào:

6

#### Dữ liệu ra:

YES

## ✓ C++ Buổi 04\_Bài 18. Bình phương nguyên tố 1.

Một số được coi là số đẹp khi nó đồng thời vừa chia hết cho một số nguyên tố và chia hết cho bình phương của số nguyên tố đó. Viết chương trình liệt kê các số đẹp như vậy trong đoạn giữa hai số nguyên dương cho trước.

### Dữ liệu vào:

2 số nguyên dương  $a, b$ . ( $1 \leq a \leq b \leq 10^6$ ).

### Dữ liệu ra:

In ra các số đẹp trong đoạn từ  $a$  tới  $b$ .

### Ví dụ:

#### Dữ liệu vào:

4 50

Copy

#### Dữ liệu ra:

4 8 9 12 16 18 20 24 25 27 28 32 36 40 44 45 48 49 50

Copy

## ✓ C++ Buổi 04\_Bài 19. Bình phương số nguyên tố 2.

Một số được coi là số đẹp khi nếu nó chia hết cho một số nguyên tố nào đó thì cũng phải chia hết cho bình phương của số nguyên tố đó. Viết chương trình liệt kê các số đẹp như vậy trong đoạn giữa hai số nguyên dương cho trước.

### Dữ liệu vào:

2 số nguyên dương  $a, b$ . ( $1 \leq a \leq b \leq 10^6$ ).

### Dữ liệu ra:

In ra các số đẹp trong đoạn từ  $a$  tới  $b$ .

### Ví dụ:

#### Dữ liệu vào:

```
4 50
```

[Copy](#)

#### Dữ liệu ra:

```
4 8 9 16 25 27 32 36 49
```

[Copy](#)

## ✔ C++ Buổi 04\_Bài 20.T-prime.

Cho số tự nhiên  $N$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các số có đúng ba ước số không vượt quá  $n$ . Ví dụ  $n=100$ , ta có các số 4, 9, 25, 49.

### Dữ liệu vào:

Số nguyên dương  $N$ . ( $1 \leq N \leq 10^{12}$ ).

### Dữ liệu ra:

Đưa ra kết quả trên một dòng.

### Ví dụ:

#### Dữ liệu vào:

100

#### Dữ liệu ra:

4 9 25 49

## ✓ C++ Buổi 04\_Bài 21.T-prime2.

Cho số tự nhiên  $N$ . Nhiệm vụ của bạn là hãy đếm tất cả các số có đúng ba ước số không vượt quá  $n$ . Ví dụ  $n=100$ , ta có các số 4 9 25 49.

### Dữ liệu vào:

Số nguyên dương  $N(1 \leq N \leq 10^{12})$ .

### Dữ liệu ra:

Đưa ra kết quả mỗi theo từng dòng.

### Ví dụ:

#### Dữ liệu vào:

100

#### Dữ liệu ra:

4

## ✓ C++ Buổi 04\_Bài 22.Số có số lượng ước là số lẻ.

Kiểm tra xem một số có số lượng ước số của nó là số lẻ

### Dữ liệu vào:

Số nguyên dương  $N$ . ( $1 \leq N \leq 2 \cdot 10^{18}$ )

### Dữ liệu ra:

In ra YES nếu  $N$  là số có số ước là số lẻ, ngược lại in NO.

### Ví dụ:

#### Dữ liệu vào:

100

#### Dữ liệu ra:

YES

## ✓ C++ Buổi 04\_Bài 23.Số Fibonacci 1.

Dãy số Fibonacci được định nghĩa như sau:  $F_1 = 0$ ,  $F_2 = 1$ ;  $F_i = F_{i-1} + F_{i-2}$ . Hãy viết chương trình in ra số Fibonacci thứ  $n$ .

### Dữ liệu vào:

Số nguyên dương  $n$ . ( $1 \leq n \leq 93$ )

### Dữ liệu ra:

Số fibonacci thứ  $n$ .

### Ví dụ:

#### Dữ liệu vào:

5

#### Dữ liệu ra:

3

## ✓ C++ Buổi 04\_Bài 24.Số Fibonacci 2.

Nhập vào một số và kiểm tra xem số vừa nhập có phải là số trong dãy fibonacci hay không?

### Dữ liệu vào:

Số nguyên không âm  $n$ . ( $0 \leq n \leq 9 \cdot 10^{18}$ )

### Dữ liệu ra:

In ra YES nếu  $n$  là số Fibonacci, ngược lại in NO.

### Ví dụ:

#### Dữ liệu vào:

6

#### Dữ liệu ra:

NO



## ✓ C++ Buổi 04\_Bài 25.Fibonacci 3.

Liệt kê những số là số nguyên tố nhỏ hơn N và có tổng các chữ số của nó là một số trong dãy số Fibonacci.

### Dữ liệu vào:

Dòng duy nhất chứa số nguyên dương N. ( $1 \leq N \leq 10000$ )

### Dữ liệu ra:

In ra các số nhỏ hơn N là số nguyên tố và thỏa mãn tổng chữ số của nó là một số trong dãy Fibonacci. Các số in cách nhau một khoảng trắng

### Ví dụ:

#### Dữ liệu vào:

150

#### Dữ liệu ra:

2 3 5 11 17 23 41 53 67 71 101 107 113 131 139

## ✓ C++ Buổi 04\_Bài 26.Tam giác Pascal.

In ra tam giác pascal với chiều cao là n.

### Input Format

Số nguyên dương n. ( $1 \leq n \leq 12$ )

### Output Format

In ra Tam giác pascal tương ứng.

### Ví dụ:

Dữ liệu vào:

5

Dữ liệu ra:

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

## ✔ C++ Buổi 04\_Bài 27.Tích các số.

Cho 1 loạt các số trong 1 dãy số,  $S$  là tích của tất cả các số trong dãy số này. Bạn hãy tìm số dư khi chia  $S$  cho  $10^9 + 7$ .

### Dữ liệu vào:

Dòng đầu tiên là  $N$  : số lượng số trong dãy số. ( $1 \leq N \leq 1000$ ) Dòng thứ 2 là  $N$  số trong dãy số. (Các số trong dãy là số không âm không vượt quá  $10^9$ )

### Dữ liệu ra:

In ra kết quả của bài toán.

### Ví dụ:

#### Dữ liệu vào:

```
6
4 8 2 3 4 5
```

#### Dữ liệu ra:

```
3840
```