

## **Buổi 4+5.**

### **Bài 1. Tìm số đẹp**

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.

INPUT

23 199

OUTPUT

1

### **Bài 2. Tìm số hoàn hảo trong đoạn cho trước.**

Viết chương trình cho phép nhập vào hai số nguyên dương và tìm tất cả các số hoàn hảo (là số có tổng các ước số nhỏ hơn nó bằng chính nó) (nếu không tồn tại số nào thì in ra 0)

INPUT

1 100

OUTPUT

6 28

**Bài 3.** Một số được gọi là số tăng giảm nếu số đó có các chữ số thỏa mãn hoặc tăng dần, hoặc giảm dần từ trái qua phải.

Hãy đếm các số nguyên tố là số tăng giảm với số chữ số cho trước

INPUT

2

OUTPUT

20

**Bài 4.** Một số được coi là đẹp nếu nó là số nguyên tố và tổng chữ số là một số trong dãy Fibonacci. Viết chương trình liệt kê trong một đoạn giữa hai số nguyên cho trước có bao nhiêu số đẹp như vậy

INPUT

2 50

OUTPUT

2 3 5 11 17 23 41

**Bài 5.** Dãy số Fibonacci được định nghĩa như sau:  $F_0 = 0$ ,  $F_1 = 1$ ;  $F_i = F_{i-1} + F_{i-2}$ . Cho số nguyên dương  $n$ , với  $2 \leq n \leq 92$ . Hãy viết chương trình in ra  $n$  số Fibonacci đầu tiên.

INPUT

5

OUTPUT

0 1 1 2 3

**Bài 6.** Viết chương trình phân tích một số nguyên thành các thừa số nguyên tố.

INPUT

28

OUTPUT

2 2 7

**Bài 7.** 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

INPUT

2 100

OUTPUT

**Bài 8.** Một số được coi là đẹp nếu nó có tính chất thuận nghịch và tổng chữ số chia hết cho 10. Bài toán đặt ra là cho trước số chữ số. Hãy đếm xem có bao nhiêu số đẹp với số chữ số như vậy

INPUT

2

OUTPUT

1

**Bài 9.** Viết chương trình cho phép nhập vào hai số nguyên dương và tìm tất cả các số nguyên tố nằm trong khoảng đó

INPUT

10 50

OUTPUT

11 13 17 19 23 29 31 37 41 43 47

**Bài 10.** Nhập một số nguyên dương N không quá 9 chữ số. Hãy đếm xem N có bao nhiêu chữ số lẻ và bao nhiêu chữ số chẵn. Nếu không tồn tại số lẻ hoặc số chẵn thì in ra kết quả là 0 cho loại số tương ứng

INPUT

12345678

OUTPUT

4 4

**Bài 11.** Viết chương trình cho phép nhập vào hai số nguyên dương và tìm tất cả các số Strong (là số có tổng giai thừa các chữ số bằng chính nó) nằm trong khoảng đó (nếu không tồn tại số nào thì in ra 0)

INPUT

1 1000

OUTPUT

1 2 145

**Bài 12.** Một số được gọi là “lộc phát” nếu chỉ có các chữ số 0,6,8. Nhập vào một số nguyên dương không quá 9 chữ số, hãy kiểm tra xem đó có phải số lộc phát hay không. Nếu đúng in ra 1, sai in ra 0

INPUT

6068

OUTPUT

1

**Bài 13.** 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 trong một đoạn giữa hai số nguyên cho trước có bao nhiêu số đẹp như vậy

INPUT

1 500

OUTPUT

161

**Bài 14.** Viết chương trình C cho phép nhập vào một số n, thực hiện tìm tích của các chữ số của n và in ra màn hình.

INPUT

1234

OUTPUT

24

**Bài 15.** Viết chương trình C cho phép nhập vào một số n và in ra tất cả các ước số của n. (Mỗi ước được liệt kê cách nhau một khoảng trắng)

INPUT

12

OUTPUT

1 2 3 4 6 12

**Bài 16.** Viết chương trình C cho phép nhập vào số n và liệt kê tất cả các số nguyên tố nhỏ hơn n. (Các số nguyên tố in ra cách nhau một khoảng trắng)

INPUT

20

OUTPUT

2 3 5 7 11 13 17 19

**Bài 17.** Viết chương trình C cho phép nhập vào n và thực hiện liệt kê các số arstrong nhỏ hơn n. (Mỗi kết quả thỏa mãn cách nhau một khoảng trắng)

INPUT

1000

OUTPUT

1 2 3 4 5 6 7 8 9 153 370 371 407

**Bài 18.** Số hoàn hảo là số có tổng các ước số (nhỏ hơn chính nó) bằng nó. Ví dụ:  $6 = 1 + 2 + 3$ . Viết chương trình C cho phép nhập vào số n và kiểm tra xem n có phải là số hoàn hảo hay không. Nếu đúng in ra 1, sai in ra 0.

INPUT

6

OUTPUT

1

**Bài 19.** Viết chương trình C cho phép nhập vào hai số a và b (a có thể lớn hơn b). Thực hiện liệt kê các số hoàn hảo nằm trong khoảng a và b.

INPUT

1 1000

OUTPUT

6 28 496

**Bài 20.** Viết chương trình C cho phép nhập vào một số và phân tích thành thừa số các số nguyên tố. (Mỗi thừa số nguyên tố cách nhau một khoảng trắng, mỗi thừa số nguyên tố chỉ liệt kê một lần)

INPUT

10

OUTPUT

2 5

**Bài 21.** Số Strong là số thỏa mãn có tổng giai thừa các chữ số của nó bằng chính nó. Ví dụ:  $145 = 1! + 4! + 5!$

Viết chương trình C cho phép nhập vào số n và kiểm tra xem n có phải số Strong hay không. Nếu đúng in ra 1 sai in ra 0.

INPUT

145

OUTPUT

1

**Bài 22.** Viết chương trình C cho phép nhập vào một số n bất kỳ, tìm và in ra chữ số đầu tiên và cuối cùng của n.

INPUT

1234

OUTPUT

1 4

**Bài 23.** Viết chương trình C cho phép nhập vào một số n, thực hiện tìm tổng các chữ số của n và in ra màn hình.

INPUT

1234

OUTPUT

10

**Bài 24.** Viết chương trình C cho phép nhập vào 1 số và kiểm tra xem số đó có phải là số nguyên tố hay không. Nếu đúng in ra 1, sai in ra 0.

INPUT

11

OUTPUT

1

**Bài 25.** Viết chương trình C cho phép nhập một số tự nhiên  $n$  và tính giai thừa của  $n$ .

INPUT

5

OUTPUT

120

**Bài 26.** Số hoàn hảo là số có tổng các ước số (nhỏ hơn chính nó) bằng nó. Ví dụ:  $6 = 1 + 2 + 3$ .

Viết chương trình cho phép nhập vào số  $n$  và liệt kê các số hoàn hảo nhỏ hơn  $n$ .

INPUT

1000

OUTPUT

6 28 496

**Bài 27.** Viết chương trình C cho phép nhập vào một số nguyên  $n$  và thực hiện đổi vị trí của chữ cái đầu tiên và chữ cái cuối cùng.

Lưu ý trong trường hợp chữ số cuối cùng là 0 thì khi đổi chỗ sẽ được loại bỏ (ví dụ 9800  $\rightarrow$  809)

INPUT

1234

OUTPUT

4231

**Bài 28.** Viết chương trình C cho phép nhập vào một số  $n$  và kiểm tra xem  $n$  có bao nhiêu chữ số.

INPUT

1234

OUTPUT

4

**Bài 29.** Số armstrong là số A có n chữ số và thỏa mãn tổng của lũy thừa bậc n của từng chữ số trong A bằng chính nó.

Ví dụ:  $371 = 3^3 + 7^3 + 1^3$

Viết chương trình C kiểm tra một số xem có phải là số armstrong hay không. Nếu đúng in ra 1, sai in ra 0.

INPUT

371

OUTPUT

1

**Bài 30.** Tam giác pascal là tam giác có công thức tính giá trị một vị trí bất kỳ như sau

$$(n,k) = n!/k!(n-k)!$$

Trong đó: n là hàng và k là cột.

Viết chương trình C cho phép nhập vào n và in ra tam giác pascal tương ứng (lưu ý mỗi phân tử cách nhau 1 khoảng trắng)

INPUT

5

OUTPUT

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

**Bài 31.**



Viết chương trình C cho phép nhập vào một số n và kiểm tra số đó có thuộc dãy fibonacci hay không. Nếu thỏa mãn in ra 1, ngược lại in ra 0.

INPUT

8

OUTPUT

1

**Bài 32.** Viết chương trình C cho phép nhập vào n. Thực hiện liệt kê các số nguyên tố nhỏ hơn n và thỏa mãn có tổng các chữ số chia hết cho 5. Có bao nhiêu số như vậy

INPUT

100

OUTPUT

5 19 23 37 41 73

6

**Bài 33.**

Viết chương trình C cho phép nhập vào số N, thực hiện liệt kê các số thuận nghịch lớn hơn 1 và nhỏ hơn N thỏa mãn không chứa chữ số 9. Có bao nhiêu số như vậy.

INPUT

100

OUTPUT

2 3 4 5 6 7 8 11 22 33 44 55 66 77 88

15

**Bài 34.** Viết chương trình C cho phép nhập vào n và liệt kê các số nguyên tố thỏa mãn nhỏ hơn n và có chữ số cuối cùng lớn nhất. Có bao nhiêu số như vậy

INPUT

200

OUTPUT

2 3 5 7 11 13 17 19 23 29 37 47 59 67 79 89 101 103 107 109 113 127 137 139 149  
157 167 179 199  
29

### **Bài 35.**

Viết chương trình C cho phép nhập vào 2 số a và b. Sắp xếp a, b theo thứ tự tăng dần tổng các chữ số.

(Nếu a và b có tổng các chữ số bằng nhau thì in ra a trước)

INPUT

99 1111

OUTPUT

1111 99

### **Bài 36.**

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.

INPUT

1 50

OUTPUT

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

**Bài 37.** Viết chương trình phân tích một số nguyên thành các thừa số nguyên tố.

INPUT

28

OUTPUT

2x2x7

**Bài 38.** Viết chương trình C cho phép nhập vào một số n, thực hiện đếm số lần xuất hiện của các chữ số nguyên tố trong n và in ra màn hình. (Liệt kê theo thứ tự xuất

hiện các chữ số)

INPUT

112345

OUTPUT

2 1

3 1

5 1

**Bài 39.** Một số được coi là số đẹp nếu nó là số nguyên tố và ngược lại của nó cũng là một số nguyên tố khác. Viết chương trình liệt kê các số đẹp như vậy trong một đoạn giữa hai số nguyên dương cho trước.

INPUT

1 50

OUTPUT

13 17 31 37

**Bài 40.** Một số được coi là số đẹp khi nếu nó chia hết cho một số nguyên tố nào đó thì cũng 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.

INPUT

1 50

OUTPUT

4 8 9 16 25 27 32 36 49