

BUỔI 5

Bài 1. Số Strong

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 2. Chữ số đầu tiên và cuối cùng

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 3. Tổng chữ số

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 4. Kiểm tra số nguyên tố

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 5. Tính giai thừa

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

INPUT

5

OUTPUT

120

Bài 6. Số hoàn hảo

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 7. Đổi chữ số đầu cuối

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 -> 809)

INPUT

1234

OUTPUT

4231

Bài 8. Đếm số chữ số

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 9. Số armstrong

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 10. Tam giác pascal

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 phần 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 11. Kiểm tra số Fibonacci

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 12. Số nguyên tố

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 13. Số thuận nghịch không chứa 9

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

1 5

Bài 14. Số nguyên tố có chữ số cuối cùng lớn nhất

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 15. Sắp xếp theo tổng chữ số

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 16. Số đẹp

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 17. Phân tích thừa số nguyên tố

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

INPUT

28

OUTPUT

2x2x7

Bài 18. Liệt kê chữ số nguyên tố

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

3112345

OUTPUT

3 2

2 1

5 1

Bài 19. Số đẹp

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 20. Số đẹp

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