

PRIME FRACTORIZATION (C PROGRAMMING)

LIỆT KÊ CÁC THỪA SỐ NGUYÊN TỐ CÓ THỂ GIỐNG NHAU

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

Input

Dòng đầu ghi số bộ test, mỗi bộ test ghi trên một dòng số nguyên dương cần phân tích (không quá 9 chữ số) .

Output

Kết quả của mỗi bộ test ghi trên một dòng, mỗi thừa số cách nhau một khoảng trống.

Ví dụ

| Input | Output |
|-------|--------|
| 2 | 2 5 |
| 10 | 2 2 5 |
| 20 | |

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb

LIỆT KÊ CÁC THỪA SỐ NGUYÊN TỐ KHÁC NHAU

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

Input

Dòng đầu ghi số bộ test, mỗi bộ test ghi trên một dòng số nguyên dương cần phân tích (không quá 18 chữ số) .

PRIME FRACTORIZATION (C PROGRAMMING)

Output

Kết quả của mỗi bộ test ghi trên một dòng, mỗi thừa số cách nhau một khoảng trống.

Ví dụ

| Input | Output |
|-------|--------|
| 2 | 2 5 |
| 10 | 2 5 |
| 20 | |

Giới hạn thời gian: 1s

Giới hạn bộ Nhớ: 65536 Kb

IN THEO MẪU PHÂN TÍCH THỪA SỐ NGUYÊN TỐ 1

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

Input

Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng số nguyên dương n không quá 9 chữ số.

Output

Mỗi bộ test viết ra thứ tự bộ test, sau đó lần lượt là các số nguyên tố khác nhau có trong tích, với mỗi số viết thêm số lượng số đó. Xem ví dụ để hiểu rõ hơn về cách viết kết quả.

PRIME FRACTORIZATION (C PROGRAMMING)

Ví dụ

| Input | Output |
|-------|------------------------|
| 3 | Test 1: 2(2) 3(1) 5(1) |
| 60 | Test 2: 2(7) |
| 128 | Test 3: 2(4) 5(4) |
| 10000 | |

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb

IN THEO MẪU PHÂN TÍCH THỪA SỐ NGUYÊN TỐ 2

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

Input

Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng số nguyên dương n không quá 9 chữ số.

Output

Mỗi bộ test viết ra kết quả phân tích theo mẫu như trong ví dụ (chú ý: giữa các số với dấu * có một khoảng trống).

Ví dụ

PRIME FRACTORIZATION (C PROGRAMMING)

| Input | Output |
|-------|------------------------|
| 3 | $60 = 2^2 * 3^1 * 5^1$ |
| 60 | $128 = 2^7$ |
| 128 | $10000 = 2^4 * 5^4$ |
| 10000 | |

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb

IN THEO MẪU PHÂN TÍCH THỪA SỐ NGUYÊN TỐ 3

Viết chương trình phân tích một số nguyên dương (không quá 6 chữ số) thành tích các thừa số nguyên tố.

Kết quả được viết theo mẫu trong Ví dụ (có chữ x giữa các thừa số)

Input:

Chỉ có một dòng ghi số n.

Output:

Ghi ra kết quả tính toán

Ví dụ:

| Input | Output |
|-------|--------|
|-------|--------|

PRIME FRACTORIZATION (C PROGRAMMING)

| | |
|----|-----------------------|
| 28 | $2 \times 2 \times 7$ |
|----|-----------------------|

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb

IN ƯỚC SỐ NGUYÊN TỐ LỚN NHẤT

Cho số nguyên dương N . Hãy đưa ra ước số nguyên tố lớn nhất của N .

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test ghi số nguyên dương N .
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq N \leq 10^{10}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

| Input: | Output: |
|--------|---------|
| 2 | 7 |
| 315 | 31 |
| 31 | |

PRIME FRACTORIZATION (C PROGRAMMING)

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb

IN ƯỚC SỐ NGUYÊN TỐ NHỎ NHẤT

Cho số nguyên dương N. Hãy đưa ra ước số nguyên tố nhỏ nhất của N.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test ghi số nguyên dương N.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq N \leq 1010$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

| Input: | Output: |
|--------|---------|
| 2 | 3 |
| 315 | 31 |
| 31 | |

Giới hạn thời gian: 1s

PRIME FRACTORIZATION (C PROGRAMMING)

Giới hạn bộ nhớ: 65536 Kb

IN 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 $28 = 2 \times 2 \times 7$.

Đầu vào

2 số **n, k**

Giới hạn

$1 \leq n, k \leq 10^9$

Đầu 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**.

Input 01

28 3

Output 01

7

CHO MẢNG, TÍNH TỔNG CÁC ƯỚC SỐ NGUYÊN TỐ PHÂN TÍCH

Cho N số nguyên. Nhiệm vụ của bạn là phân tích các số nguyên đã cho dưới dạng tích của các thừa số nguyên tố, sau đó tính tổng các ước số nguyên tố này.

Input:

- Dòng đầu tiên số nguyên N ($1 \leq N \leq 106$).
- N dòng tiếp theo, mỗi dòng gồm một số nguyên có giá trị không vượt quá $2 \cdot 106$.

Output:

In ra một số nguyên là đáp án tìm được.

Ví dụ:

PRIME FRACTORIZATION (C PROGRAMMING)

| Input: | Output: |
|--------|---------|
| 5 | 47 |
| 7 | |
| 9 | |
| 10 | |
| 13 | |
| 100 | |

viết hàm sàng

viết hàm phân tích nguyên tố

Giải thích test:

$$7 = 7$$

sum += ptnt(a[i])

$$9 = 3 \times 3 \rightarrow 3 + 3 = 6$$

$$10 = 2 \times 5 \rightarrow 2 + 5 = 7$$

$$13 = 13$$

$$100 = 2 \times 2 \times 5 \times 5 \rightarrow 2+2+5+5 = 14$$

$$\text{Cộng lại, } 7 + 6 + 7 + 13 + 14 = 47.$$

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 65536 Kb

TÍCH CÁC THỪA SỐ NGUYÊN TỐ KHÁC NHAU

PRIME FRACTORIZATION (C PROGRAMMING)

Cho một số nguyên dương không quá 9 chữ số. Người ta phân tích số đó thành tích các thừa số nguyên tố sau đó tính lại một giá trị mới bằng cách nhân các thừa số nguyên tố khác nhau của số đó.

Ví dụ: Số 72 được phân tích thành $2^3 * 3^2$. Giá trị tính được sẽ là $2 * 3 = 6$

Dữ liệu vào

- Dòng đầu ghi số bộ test, không quá 10
- Mỗi bộ test là một số nguyên dương không quá 109

Kết quả

- Với mỗi bộ test, ghi ra kết quả tính được.

Ví dụ

| Input | Output |
|-------|--------|
| 3 | 6 |
| 72 | 10 |
| 1000 | 997 |
| 997 | |

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 65536 Kb