**ĐỀ SỐ 36 – Ngày 20 - 10 - 2022**

**Bài 1. Số đặc biệt**

Khôi có một mảng số tự nhiên A có N phần tử. Anh ấy phải tìm ra tất cả các số đặc biệt K.

Biết rằng số đặc biệt K phải thỏa mãn những điều sau

1) K>1

2) A[1]%K = A[2]%K = A[3]%K = ... = A[N]%K

Hãy giúp Khôi tìm ra tất cả các số đặc biệt K.

**Dữ ệu vào**

* Dòng đầu tiên chứa 1 số nguyên dương N(2≤N≤10^5)
* Gồm N dòng, dòng ii chứa giá trị của Ai(1≤Ai≤10^9)
* Các số trong mảng A khác nhau đôi một

*Dữ liệu Input đảm bảo có ít nhất*1*số*K*thỏa mãn và nhiều nhất* 106*số*K*thỏa mãn*

**Kết quả**

* Tất cả các số đăc biệt K theo thứ tự tăng dần. (Mỗi số trên 1 dòng)

Ví dụ:

|  |  |
| --- | --- |
| **sodb.inp** | **sodb.out** |
| 3  38  6  34 | 2  4 |

**Bài 2. SỐ ƯỚC**

Cho số nguyên dương *N*. Giai thừa của *N*, kí hiệu là *N*!, là tích của các số tự nhiên từ 1 đến *N*. Gọi T là số lượng ước lớn hơn 1 của *N*!. Ví dụ với *N* = 4, ta có 4! = 24. Như vậy 4! có 7 ước lớn hơn 1 là: 2, 3, 4, 6, 8, 12, 24.

**Yêu cầu**: Cho *N*, hãy xác định *T*.

**Dữ liệu**: Vào từ file văn bản **DIVISORS.INP** trong đó chứa duy nhất số *N* (*N*20, trong đó 50% số test có *N*10).

**Kết quả**: Ghi ra file văn bản **DIVISORS.OUT** số *T* tìm được.

**Ví dụ**:

|  |  |  |
| --- | --- | --- |
| **DIVISORS.INP** |  | **DIVISORS.OUT** |
| 4 |  | 7 |

**Bài 3. Số nguồn**

Xét một số N nguyên dương, Ví dụ: N = 836, xét tổng M = 836 + 8 + 3 +6 = 853, khi đó ta gọi số N = 836 là số nguồn của số M = 853. Có những số có thể có nhiều nguồn, ví dụ số: M = 909 có hai số nguồn là 891 và 900.

Cho trước số M (M ≤ 2\*109). Nhiệm vụ của bạn là tìm số nguồn của M, nếu như M có nhiều nguồn thì đưa ra kết quả là số nguồn nhỏ nhất.

Input

* Số nguyên dương M

Output

* Số nguồn nhỏ nhất của số M, trường hợp không tìm được số nguồn của M thì đưa ra -1.

Ví dụ

|  |  |
| --- | --- |
| **Songuon.Inp** | **Songuon.Out** |
| 21 | 15 |
| 909 | 891 |
| 3 | -1 |

* Chú ý: 50% test M ≤ 105

**Bài 4.** **Khóa số**

Các nhà khảo cổ học đã phát hiện ra một số kho báu bí mật được các vua Hùng xây dựng rất kiên cố và không thể phá bỏ. Họ cho rằng trong đó có thể là những khối tài sản về lịch sử và văn hóa rất có giá trị và họ tìm cách mở cánh cửa của những kho báu đó.

|  |  |  |
| --- | --- | --- |
| Trên cửa mỗi kho báu có một bảng gồm 2 hàng, hàng 1 đã ghi sẵn số nguyên dương N (≤106), hàng 2 chứa 2 khoá số K1 và K2 như hình bên. | N | |
| K1 | K2 |

Trong khi khảo sát, các nhà khảo cổ đã phát hiện một phiến đá có ghi cách để mở khoá như sau: cửa có bảng chứa số N sẽ tương ứng với K1 và K2 là:

* Điều chỉnh khoá số K1 về số bằng số lượng ước nguyên tố của N;
* Điều chỉnh khoá số K2 về số bằng tổng các ước nguyên tố của N thì cánh cửa sẽ tự động mở ra và nhà khảo cổ có thể vào bên trong kho báu một cách dễ dàng.

Ví dụ: Ở nhà kho trên cửa ghi số N=12, có các ước của N là 1, 2, 3, 4, 6, 12 chỉ có 2 ước nguyên tố là 2 và 3 nên K1=2 và K2=5.

**Dữ liệu:** vào từ filevăn bản KHOASO.INPchứa duy nhất một số nguyên dương *N;*

**Kết quả:** Ghi ra filevăn bản KHOASO.OUT hai số nguyên K1 và K2.

**Ví dụ:**

|  |  |
| --- | --- |
| **KHOASO.INP** | **KHOASO.OUT** |
| 12 | 2 5 |

***Ràng buộc:***

* Có 30% số các test ứng với 30% số điểm của bài có ;
* Có 40% số test khác ứng với 40% số điểm của bài có ;
* Có 30% số test còn lại ứng với 30% số điểm của bài có.

# Bài 5. Số thứ k

Đức viết các số tự nhiên từ 2, 3, 4, …, *n* trên mặt bàn và tiến hành xóa các số đó như sau:

Chọn số nhỏ nhất chưa xóa (gọi là số x), xóa số x và tất cả các số chưa xóa là bội của x (thứ tự các số được xóa từ số nhỏ đến số lớn) và cứ tiếp tục quay lại như vậy.

**Yêu cầu:** Tìm xem, số thứ *k* được xóa là số nào (*k* < *n <* 107).

**Dữ liệu** cho trong file **KDELE.INP** gồm hai số nguyên dương *n* và *k* được ghi trên một dòng.

**Kết quả** ghi ra file **KDELE.OUT** là số thứ *k* được xóa.

*Ví dụ*:

|  |  |
| --- | --- |
| **KDELE.INP** | **KDELE.OUT** |
| 6 1 | 2 |
| 6 5 | 5 |

## **Bài 6. Số đối xứng**

Một số tự nhiên được gọi là một số đối xứng nếu đọc số đó từ trái sang phải cũng giống như đọc số đó từ phải sang trái. Ví dụ, các số: 1, 121, 101, 3443 là các số đối xứng, các số 123, 332 là các số không đối xứng.

**Yêu cầu**: Cho số tự nhiên *n*. Tìm số đối xứng lớn nhất không lớn hơn *n*.

**Dữ liệu** cho trong file **NUMPALIN.INP** gồm nhiều dòng, mỗi dòng ghi một số nguyên dương *n*. **Kết quả** ghi ra file **NUMPALIN.OUT** gồm các dòng tương ứng là kết quả tìm được, mỗi dòng gồm số x là số đối xứng lớn nhất không lớn hơn *n*.

*Ví dụ*:

|  |  |
| --- | --- |
| **NUMPALIN.INP** | **NUMPALIN.OUT** |
| 2  10  123  1009  2001 | 2  9  121  1001  1991 |

**Sub1:** 1 ≤ *n* ≤ 106

**Sub2:** 106 ≤ *n* ≤ 101000.

# Bài 7. Tập chia hết

Cho dãy số nguyên dương *a1, a2, …, an*. Hãy chọn tập *S* gồm nhiều số hạng nhất thỏa mãn: ∀x ∈S thì x chia hết cho y, ∀y ∈S, y < x Ví dụ: Dãy 1, 4, 2, 6, 8, 7, 2. Tập *S* = {1, 4, 2, 8, 2}.

**Dữ liệu** cho trong file **DIVISOR.INP** như sau:

* Dòng đầu ghi số nguyên dương *n* (*n* ≤ 1000).
* Dòng thứ hai ghi *n* số nguyên dương *a1, a2, … , an* (*ai ≤* 109).

**Kết quả** ghi ra file **DIVISOR.OUT** gồm một số là số số hạng của tập S.

*Ví dụ:*

|  |  |
| --- | --- |
| **DIVISOR.INP** | **DIVISOR.OUT** |
| 7  1 4 2 6 8 7 2 | 5 |

# Bài 8. BITSTRING

Ta gọi xâu nhị phân độ dài *n* là dãy gồm *n* kí hiệu, mỗi kí hiệu chỉ là 1 hoặc 0. Như đã biết, có tất cả 2*n* xâu như vậy. Trong bài toán này, ta chỉ quan tâm đến những xâu nhị phân độ dài *n* chứa đoạn gồm *k* số 1 liên tiếp. Vì con số này là rất lớn khi *n* và *k* lớn, nên chỉ cần đưa ra phần dư trong phép chia của số này cho 1000007.

**Yêu cầu**: Cho trước hai số nguyên dương *n* và *k*, hãy tìm phần dư trong phép chia của số lượng xâu

nhị phân độ dài *n* chứa đoạn gồm *k* số 1 liên tiếp cho 1000007.

**Dữ liệu** cho trong file **BITSTRING.INP** gồm hai số nguyên dương *n* và *k* (1 ≤ *n* ≤ 105, 1 ≤ *k* ≤ 100).

**Kết quả** ghi ra file **BITSTRING.OUT** là phần dư trong phép chia của số lượng xâu nhị phân độ dài *n* chứa đoạn gồm *k* số 1 liên tiếp cho 1000007.

*Ví dụ:*

|  |  |  |
| --- | --- | --- |
| **BITSTRING.INP** | **BITSTRING.OUT** | **Giải thích** |
| 3 2 | 3 | 110, 011, 111 |
| 5 1 | 31 | Có 25 xâu, trừ 1 xâu 00000 |
| 5 6 | 0 | *k* > *n* |