**varMỘT SỐ BÀI TOÁN VỀ SỐ HỌC**

**(DÙNG CÂU LỆNH LẶP)**

**Bài 1: Tổng các chữ số**

Viết chương trình nhập vào số nguyên dương N và tính tổng các chữ số là số lẻ của số N?

***Dữ liệu vào trong file văn bản “Bai1.inp” có dạng:***Một dòng duy nhất chứa số N *(N tối đa 100 chữ số)* .

***Kết quả cho ra file văn bản “Bai1.out” có dạng:***Một dòng duy nhất chứa tổng các số là số lẻ của số N.

***Ví dụ:***

|  |  |
| --- | --- |
| **Bai1.inp** | **Bai1.out** |
| 29475 | 21 |

Bài 2: Viết chương trình nhập vào số tự nhiên N rồi thông báo lên màn hình số đó có phải là số nguyên tố hay không.

Bài 3: Viết chương trình nhập vào số nguyên N. In ra màn hình tất cả các ước số của N.

Bài 4: Viết chương trình tìm USCLN và BSCNN của 2 số a, b được nhập vào từ bàn phím.

**Bài5**: Tìm tất cả các chữ số có ba chữ số abc sao cho tổng các lập phương của các chữ

số thì bằng chính số đó ( abc = a3 + b3 + c3).

**Bài 6** Nhập vào một số nguyên dương n. N có phải số hoàn hảo không.

**Bài 7:a,**Tìm các số tự nhiên nhỏ hơn hoặc bằng n mà sau khi làm phép phân tích ra thừa số nguyên tố có nhiều nhân tử nhất.

Ví dụ n=9 . Các số có nhiều nhân tử nhất sau khi làm phép phân tích là: 8 = 2.2.2

B .Viết chương trình cho phép phân tích một số ra thừa số nguyên tố và ghi kết quả dưới dạng tích các lũy thừa. Ví dụ: 300 = 2^2.3.5^2

**Bài 8:** Mọi số tự nhiên đều có thể viết được dưới dạng tổng của hai số nguyên tố. Viết chương trình thực hiện tách một số tự nhiên thành tổng của hai số nguyên tố.

**Bài 9:** Hai số tự nhiên A, B được coi là hữu nghị nếu như số này bằng tổng các ước số thực sự của số kia và ngược lại. Lập trình tìm đưa ra các cặp số hữu nghị trong phạm vi từ 1 đến 10000. (Lưu ý: số 1 được coi là ước số của mọi số còn mỗi số không được coi là ước số của chính nó).

Bài toán: Tính N giai thừa (n<200) sau đó đưa ra số cuối cùng khác 0 của N giai thừa.

*Bài 10*: Cho số tự nhiên n>0, ta viết n dưới n dạng n= 2m+p, với m là số tự nhiên lớn nhất. Ví dụ số tự nhiên n=11 được viết dưới dạng n= 23+3 trong đó m=3, p = 3; Yêu cầu: Viết chương trình nhập vào số tự nhiên n. Xuất ra màn hình giá trị của m và p.

Bài 11: Bài toán: số rõ ràng

Số nguyên dương n là số rõ ràng khi tạo một số mới bằng cách lấy tổng bình phương các chữ số của nó, với số mới này ta lại lặp lại công việc trên. Nếu trong quá trình tạo ta nhận được số mới là 1 thì số N ban đầu là số rõ ràng

Ví dụ: n=19->82(12+92) ->68->100->1 (19 là số rõ ràng)

N=12->5->,,,,,,,,,,,->145 (12 khong phải)

Cho số nguyên dương N. tìm số rõ ràng nhỏ nhất >n nhất

Bài 12: Bài toán: SỐ MERSEN

Số nguyên N được gọi là số Mersen nếu nó biểu diễn được dưới dạng

N= 2p + 1 , với P là số nguyên tố

Nhập vào số N, tìm tất cả các số Mersen ≤ N

Bài 13: Bài toán: SỐ NGUYÊN TỐ TƯƠNG ĐƯƠNG

Hai số nguyên được gọi là nguyên tố tương đương nếu có chung các ước nguyên tố.

VD: 75 và 15 là nguyên tố tương đương vì có chung các ước nguyên tố 3 và 5. Cho số tự nhiên N, tìm số lớn nhất <N và nguyên tố tương đương với N

(tương tự tìm số nhỏ nhất > N và nguyên tố tương đương với N)

(Số lớn nhất <N và nguyên tố tương đương với N)

Bài 14: Bài toán : TỔNG TẬN CÙNG CỦA m CHỮ SỐ

Nhập vào 1 số tự nhiên n và nhập vào m, sau đó tính tổng m các số tận cùng của n.  
vd: n = 365 m =2 tổng = 5+6=11.

**Bài 15:** : SỐ PHẢN NGUYÊN TỐ: Một số tự nhiên n được gọi là số phản nguyên tố nếu nó ó nhiều ước số nhất trong n số tự nhiên đầu tiênYêu cầu: Cho số K (K<=10000) ghi ra số phản nguyên tố lớn nhất nhỏ hơn hoặc bằng K.

**Bài 16:**  Nhập vào từ bàn phím một số N nguyên dương (N<=5000)

*Câu a)* Hãy phân tích N thành tổng của hai số nguyên tố (nếu được) và thông báo không được nếu không

**Bài 17:** Số nguyên tố rút gọn của một số tự nhiên n chính là tổng các ước nguyên tố của n.

Ví dụ: n=252=2.2.3.3.7 (n có 3 ước nguyên tố là 2, 3 và 7)

Số nguyên tố rút gọn của n là 2+3+7=12

Yêu cầu: Nhập số tự nhiên n từ bàn phím, in ra số nguyên tố rút gọn của n. (1<n<1000000)

**Bài 18: Đếm số Fibonaci**

Cho dãy số A có N phần tử nguyên 2 byte. Viết chương trình đếm xem trong dãy A có bao nhiêu số Fibonaci (F) khác nhau, biết rằng: F1 = 1; F2 = 1; Fi = Fi-1­­ + Fi-2, *với i>=3*.

***Dữ liệu vào trong file văn bản “Bai3.inp” có dạng:***

- Dòng đầu chứa số N *(1<N<=10.000)*

- Dòng thứ hai chứa các phần tử của dãy A *(mỗi số cách nhau một dấu cách)*

***Kết quả cho ra file văn bản “Bai3.out” có dạng:***

Một dòng duy nhất chứa số lượng số Fibonaci khác nhau có trong dãy A.

***Ví dụ:***

|  |  |
| --- | --- |
| **Bai3.inp** | **Bai3.out** |
| 10  5 4 7 3 2 8 1 9 1 5 | 5 |

Bài 19: Bài toán: Số đơn điệu

Định nghĩa: Các số nguyên dương luân phiên tăng giảm hoặc giảm tăng được gọi là các số đơn điệu.

Ví dụ: 4537 vì 4 < 5 > 3 < 7 //

Yêu cầu: Viết chương trình xác định số chữ số đầu tiên lớn nhất tạo thành số đơn điệu của một số cho trước.

Ví dụ: Dulieu.Vao 4537

KetQua.Ra 4

**Bài 20:** Một số có tổng các ước nhỏ hơn nó bằng chính nó được gọi là số hoàn chỉnh.

Ví dụ: 6 có các ước nhỏ hơn nó là 1, 2, 3. Tổng là 1 + 2 + 3 = 6.

Viết chương trình xét xem một số n được nhập từ bàn phím có phải là số hoàn chỉnh không.

*Bài 21 - Siêu nguyên tố*

Số siêu nguyên tố là số nguyên tố mà khi bỏ một số tuỳ ý các chữ số bên phải của nó thì phần còn lại vẫn tạo thành một số nguyên tố.

Ví dụ 7331 là một số siêu nguyên tố có 4 chữ số vì 733, 73, 7 cũng là các số nguyên tố.

Viết chương trình nhập dữ liệu vào là một số nguyên N (0< N <10) và đưa ra kết quả là các số siêu nguyên tố có N chữ số cùng số lượng của chúng.

Bài 22 : Số nguyên dương n gọi là song tố khi n là số nguyên tố và tổng chữ số của nó cũng là số nguyên tố.

Yêu cầu: Viết chưong trình thực hiện công việc sau:

Nhập vào số nguyên n. Hãy cho biết n có là số song tố hay không?

*Ví dụ:*

*N=11 n là song tố;*

Bài 23: Bài toán: Số nguồn của số M

Giả thiết N là số nguyên dương, số nguyên M là tổng của N với các chữ số của nó do đó N là nguyền của số nguyên M

Vd: N=245 khi đó m=245+2+4+5=256 => 256 là nguồn của 245

Bài 24: chữ số thứ k

Viết chương trình nhập hai số tự nhiên N và k. Hãy cho biết chữ số thứ k tính từ trái sang phải và từ phải về trái trong số N là số mấy? nếu k lớn hơn độ dài của N hoặc k bằng 0 thì thông báo không tìm được.

Ví dụ 1: Với N và k được nhập: N = 65932, k = 4

Kết quả tìm được là 3.

Ví dụ 2: Với N và k được nhập: N = 65932, k = 10

Kết quả tìm được là -1 ( k lớn hơn độ dài số N).

**Bài 25: Với mỗi số nguyên dương a, ta gọi số đồng dạng với a là số nguyên dương thu được từ a bằng cách sắp xếp theo thứ tự không tăng các chữ số trong cách viết a dưới dạng hệ đếm thập phân.**

**Ví dụ: nếu a=6334 thì số đồng dạng với nó là 6433, còn nếu a=374 thì số đồng dạng với nó là 743.**

**Bài 26: Cho a, b là hai số nguyên dương. Ta gọi tổng đồng dạng của a và b là số đồng dạng với tổng của số đồng dạng với a và số đồng dạng với b.**

**Vi dụ: a=6334 và b=374 thì tổng của số đồng dạng với a và số đồng dạng với b là 6433+743=7176. vì thế tổng đồng dạng của 6334 và 374 là 7761.**

**Y/c: cho hai số a và b hãy tính tổng đồng dạng của chúng.**

**Bài 27:** Viết chương trình cho nhập hai số tự nhiên N và k. Hãy cho biết chữ số thứ k tính từ trái sang phải trong số N là số mấy? nếu k lớn hơn độ dài của N hoặc k bằng 0 thì thông báo không tìm được.

**Bài 28: Bài toán**: số tương lai  
Viết chương trình kiểm tra xem k có phải là số “tương lai” hay không (số “tương lai” là số có các ước (khác 1) là các số nguyên tố).  
VD: số 10 có ước là 2 và 5 là các số nguyên tố nên 10 là số “tương lai”.  
Bài 29: Bài toán:

Năm 1973, nhà Toán học Neil Sloan đưa ra khái niệm độ bền của một số nguyên không âm N như sau:

* Nếu N có một chữ số thì độ bền của N bằng 0.
* Nếu N có từ 2 chữ số trở lên thì độ bền của N bằng độ bền của số nguyên là tích các chữ số của N cộng 1.

Cho N, tìm số bé hơn N có độ bền lớn nhất (0 ≤ N ≤ 2.000.000.000).

*Ví dụ*

|  |  |  |
| --- | --- | --- |
| Persist.inp | persist.out | Giải thích |
| 100 | 77 | Doben(77)=Doben(49)+1=Doben(36)+1+1=Doben(18)+1+1+1=Doben(8)+1+1+1+1=0+1+1+1+1=4 |

**Bài 30** Viết chương trình cho phép cộng hai phân số.

Bài 31. Nhân hai phân số

Cho 2 phân số A/B và C/D.

Hãy xác định 2 số nguyên dương E và F thỏa mãn 2 điều kiện sau:

+ Diều kiện 1: E/F = A/B x C/D

+ Điều kiện 2: E/F là phân số tối giản

Bài 32: Bài toán: xóa số vòng tròn thứ p

Cho N là số nguyên dương từ 1 đến N sắp xếp theo vòng tròn theo chiều kim đồng hồ, cho p là số nguyên dương p<=N. từ vị trí thứ p sau đó bỏ qua 3 số, xóa số thứ tư theo chiều kim đồng hồ quá trình tiếp diễn như thế cho tới khi còn 1 số, hỏi số đó là số nào?

Bài 33: Bài xóa số vòng tròng

Viết các số tự nhiên từ 1 đến 2009 theo vòng tròn theo chiều kim đồng hồ, cũng theo chiều đó bắt đầu từ 1 cứ đếm từ 1 đến 612 thì xóa số đó đi, lại bắt đầu từ số còn lại đứng sau số vừa bị xóa, lặp lại quá trình đến khi còn một số thì dừng, a/ hỏi số đó là số nào?

b/ muốn số còn lại là số thứ L (0<l<210) thì ban đầu cần xuất phát từ số nào?

Bài 10:Bài toán: Số đẹp

Người ta định nghĩa một số nguyên dương N là số đẹp nếu N thỏa mãn 1 trong hai điều kiện sau N bằng 9

Gọi F(N) là tổng các chữ số của N thì F(N) cũng là số đẹp

Cho số nguyên dương N (N<10100 ) hãy kiểm tra xem N có là số đẹp hay không (hoặc liệt kê các số đẹp rất lớn)

Bài 34: Bài toán: “Số rắn hai đầu”

“Số rắn hai đầu” là một số nguyên dương N sao cho: khi thêm hai chữ số a, b vào hai đầu số N ấy (theo dạng ) sẽ được số mới có giá trị nâng lên 99 lần. Tức là:  
Cho trước hai chữ số nguyên a, b (0 ≤ a, b ≤ 9). Tìm “Số rắn hai đầu” N?  
Tổ chức dữ liệu:  
• Dữ liệu vào: từ tập tin Serpent.inp gồm 2 số a, b cách nhau một dấu cách.  
• Dữ liệu ra: là tập tin Serpent.out chứa số N cần tìm. Trong trường hợp có nhiều hơn một số N thỏa mãn, hãy đưa ra số bé nhất.  
Cho biết với dữ liệu vào, luôn tồn tại kết quả.