

[Hàm, Lý Thuyết Số]. Luyện tập viết hàm 1 (Quan trọng)

Problem

Submissions

Discussions

Cho số nguyên n không âm. Viết hàm xử lý các yêu cầu sau

- 1. Kiểm tra n là số nguyên tố, nếu đúng in 1, sai in 0.
- 2. In tổng chữ số của n .
- 3. In tổng chữ số chẵn của n .
- 4. In tổng chữ số của n là số nguyên tố.
- 5. In số lật ngược của n . Ví dụ 123 in 321.
- 6. In số lượng ước của n là số nguyên tố (làm tương tự như phân tích thừa số ng tố).
- 7. In ước nguyên tố lớn nhất của n (làm tương tự như phân tích thừa số ng tố).
- 8. Kiểm tra nếu n tồn tại ít nhất 1 số 6, nếu đúng in 1, sai in 0.
- 9. Kiểm tra nếu tổng chữ số của n chia hết cho 8, nếu đúng in 1, sai in 0.
- 10. Tính tổng giai thừa các chữ số của n và in ra. Ví dụ $n = 123$, tổng $= 1! + 2! + 3!$
- 11. Kiểm tra n có phải chỉ được tạo bởi 1 số hay không? Ví dụ 222, 333, 99999. Đúng in ra 1, sai in ra 0.
- 12. Kiểm tra n có phải có chữ số tận cùng là lớn nhất hay không, tức là không có chữ số nào của n lớn hơn chữ số hàng đơn vị của nó. nếu đúng in 1, sai in 0.
- 13. In tổng lũy thừa chữ số của n với số mũ là số chữ số. ví dụ 123 thì tính $1^3 + 2^3 + 3^3$.

Input Format

Số duy nhất n

Constraints

$2 \leq n \leq 1000$;

Output Format

In ra 13 dòng tương ứng với các yêu cầu ở trên.

Sample Input 0

```
36
```

Sample Output 0

```
0
9
6
3
63
2
3
1
0
726
0
1
45
```

```

#include <bits/stdc++.h>
using namespace std;

using ll = long long;

bool ham1(int n){
    if(n < 2) return false;
    for(int i = 2; i <= sqrt(n); i++){
        if(n % i == 0){
            return false;
        }
    }
    return true;
}

int ham2(int n){
    int sum = 0;
    while(n != 0){
        sum += n % 10;
        n /= 10;
    }
    return sum;
}

int ham3(int n){
    int sum = 0;
    while(n != 0){
        if(n % 10 % 2 == 0){
            sum += n % 10;
        }
        n /= 10;
    }
    return sum;
}

int ham4(int n){
    int sum = 0;
    while(n != 0){
        int r = n % 10;
        if(r == 2 || r == 3 || r == 5 || r == 7)
            sum += r;
        n /= 10;
    }
    return sum;
}

int ham5(int n){
    int rev = 0;
    while(n){
        rev = rev * 10 + n % 10;
        n /= 10;
    }
    return rev;
}

```

```
int ham6(int n){
    int dem = 0;
    for(int i = 2; i <= sqrt(n); i++){
        if(n % i == 0){
            ++dem;
            while(n % i == 0){
                n /= i;
            }
        }
    }
    if(n > 1)
        ++dem;
    return dem;
}
//60 = 2 2 3 5
//28 = 2 2 7
//13 = 13
//100 = 2 2 5 5

int ham7(int n){
    int ans;
    for(int i = 2; i <= sqrt(n); i++){
        if(n % i == 0){
            ans = i;
            while(n % i == 0){
                n /= i;
            }
        }
    }
    if(n > 1)
        ans = n;
    return ans;
}

int ham8(int n){
    while(n){
        if(n % 10 == 6)
            return 1;
        n /= 10;
    }
    return 0;
}
```

```
int ham9(int n){
    int sum = 0;
    while(n){
        sum += n % 10;
        n /= 10;
    }
    if(sum % 8 == 0) return 1;
    else return 0;
}

int gt(int n){
    int res = 1;
    for(int i = 1; i <= n; i++)
        res *= i;
    return res;
}

int ham10(int n){
    int sum = 0;
    while(n){
        sum += gt(n % 10);
        n /= 10;
    }
    return sum;
}

bool ham11(int n){
    int r = n % 10;
    while(n){
        if(n % 10 != r)
            return false;
        n /= 10;
    }
    return true;
}

bool ham12(int n){
    int r = n % 10;
    while(n){
        if(n % 10 > r)
            return false;
        n /= 10;
    }
    return true;
}

int count(int n){
    int dem = 0;
    while(n){
        ++dem;
        n /= 10;
    }
    return dem;
}
```

```
int ham13(int n){  
    int sum = 0;  
    int d = count(n);  
    while(n){  
        sum += pow(n % 10, d);  
        n /= 10;  
    }  
    return sum;  
}
```

```
int main(){  
    int n; cin >> n;  
    cout << ham1(n) << endl;  
    cout << ham2(n) << endl;  
    cout << ham3(n) << endl;  
    cout << ham4(n) << endl;  
    cout << ham5(n) << endl;  
    cout << ham6(n) << endl;  
    cout << ham7(n) << endl;  
    cout << ham8(n) << endl;  
    cout << ham9(n) << endl;  
    cout << ham10(n) << endl;  
    cout << ham11(n) << endl;  
    cout << ham12(n) << endl;  
    cout << ham13(n) << endl;  
}
```

[Hàm, Lý Thuyết Số]. Luyện tập viết hàm 2 (Quan trọng)

Problem

Submissions

Discussions

Cho số nguyên không âm N , bạn hãy viết 10 hàm tương ứng để in ra kết quả, in ra 1 hoặc 0 cho mỗi yêu cầu (Mã nguồn tham khảo Java : <https://ideone.com/t8Gcmd>)

- 1. Số lượng chữ số chẵn của n là một số lẻ
- 2. N có số lượng chữ số chẵn $>$ số lượng chữ số lẻ
- 3. N có chữ số đầu bằng chữ số cuối
- 4. Tổng chữ số của N có tận cùng là 8
- 5. Tổng chữ số của N là số nguyên tố
- 6. Các chữ số đứng cạnh nhau của n chênh lệch nhau đúng 1 đơn vị (số có 1 chữ số thỏa mãn)
- 7. N có chữ số đầu tiên lớn hơn tất cả các chữ số còn lại của nó, (số có 1 chữ số thỏa mãn)
- 8. N có tổng chữ số là một số trong dãy fibonacci
- 9. N có tổng chữ số là một số thuận nghịch
- 10. N chỉ bao gồm các chữ số 0, 6 hoặc 8

Input Format

- Dòng 1 là T : số bộ test
- T dòng tiếp theo mỗi dòng là số tự nhiên N

Constraints

- $1 \leq T \leq 1000$
- $0 \leq N \leq 10^{18}$

Output Format

- Đối với mỗi test in ra 10 dòng tương ứng với kết quả của 10 yêu cầu trên, mỗi test cách nhau 1 dòng trống

Sample Input 0

```
1
87654418
```

Sample Output 0

```
1
1
1
0
1
0
0
0
0
0
```



```

#include <bits/stdc++.h>
using namespace std;

using ll = long long;

bool nt(ll n){
    for(int i = 2; i <= sqrt(n); i++){
        if(n % i == 0) return false;
    }
    return n > 1;
}

int ham1(ll n){
    if(n == 0) return 1;
    int dem = 0;
    while(n){
        if(n % 2 == 0) ++dem;
        n /= 10;
    }
    return dem % 2 == 1;
}

int ham2(ll n){
    if(n == 0) return 1;
    int chan = 0, le = 0;
    while(n){
        if(n % 2 == 0) ++chan;
        else ++le;
        n /= 10;
    }
    return chan > le;
}

int ham3(ll n){
    int last = n % 10;
    while(n >= 10){
        n /= 10;
    }
    return n == last;
}

int ham4(ll n){
    int tong = 0;
    while(n){
        tong += n % 10;
        n /= 10;
    }
    return tong % 10 == 8;
}

```

```

int ham5(ll n){
    int tong = 0;
    while(n){
        tong += n % 10;
        n /= 10;
    }
    return nt(tong);
}

int ham6(ll n){
    while(n >= 10){
        if(abs(n % 10 - (n / 10) % 10) != 1)
            return 0;
        n /= 10;
    }
    return 1;
}

int ham7(ll n){
    int max_cur = n % 10;
    while(n >= 10){
        if(n % 10 > max_cur){
            max_cur = n % 10;
        }
        n /= 10;
    }
    return n > max_cur;
}

bool fibo(long long n){
    if(n == 0 || n == 1)
        return true;
    //sinh ra fibonacci va so sanh voi n
    long long fn1 = 1, fn2 = 0;
    for(int i = 2; i <= 92; i++){
        long long fn = fn1 + fn2; // so fibonacci thu i
        cout << i << " " << fn << endl;
        if(fn == n)
            return true;
        fn2 = fn1;
        fn1 = fn;
    }
    return false;
}

```

```

int ham8(ll n){
    int tong = 0;
    while(n){
        tong += n % 10;
        n /= 10;
    }
    return fibo(tong);
}

int tn(ll n){
    ll rev = 0, tmp = n;
    while(n){
        rev = rev * 10 + n % 10;
        n /= 10;
    }
    return tmp == rev;
}

int ham9(ll n){
    int tong = 0;
    while(n){
        tong += n % 10;
        n /= 10;
    }
    return tn(tong);
}

int ham10(ll n){
    while(n){
        int r = n % 10;
        if(r != 0 && r != 6 && r != 8) return 0;
        n /= 10;
    }
    return 1;
}

int main(){
    int t; cin >> t;
    while(t--){
        ll n;
        cin >> n;
        cout << ham1(n) << endl;
        cout << ham2(n) << endl;
        cout << ham3(n) << endl;
        cout << ham4(n) << endl;
        cout << ham5(n) << endl;
        cout << ham6(n) << endl;
        cout << ham7(n) << endl;
        cout << ham8(n) << endl;
        cout << ham9(n) << endl;
        cout << ham10(n) << endl;
        cout << endl;
    }
}

```

[Hàm, Lý Thuyết Số]. Bài 1. Kiểm Tra Số Nguyên Tố

Problem

Submissions

Discussions

Kiểm tra một số nguyên không âm N có phải là số nguyên tố hay không?

Input Format

Dòng duy nhất chứa số nguyên dương N

Constraints

$$0 \leq N \leq 10^9$$

Output Format

In ra YES nếu n là số nguyên tố, ngược lại in NO.

Sample Input 0

```
999999999
```

Sample Output 0

```
NO
```

Sample Input 1

```
17
```

Sample Output 1

```
YES
```

[Hàm, Lý Thuyết Số]. Bài 3. Kiểm Tra Số Nguyên Tố Với Nhiều Test

Problem

Submissions

Discussions

Ở bài tập này yêu cầu bạn kiểm tra số nguyên tố với nhiều trường hợp khác nhau.

Input Format

Dòng đầu tiên là số lượng test case T ; Mỗi test case là một số nguyên n

Constraints

$1 \leq T \leq 1000$; $0 \leq n \leq 10^6$

Output Format

In ra kết quả mỗi test case trên một dòng. In YES nếu n là số nguyên tố, ngược lại in NO.

Sample Input 0

```
20
364
12401
4152
4624
12783
1868
14521
24213
4740
19037
6992
9390
8929
27797
18685
13291
11424
10292
3534
30641
```

Sample Output 0

```
NO
YES
NO
NO
NO
NO
NO
NO
NO
YES
NO
NO
YES
NO
NO
YES
NO
NO
NO
NO
```

[Hàm, Lý Thuyết Số]. Bài 4. Số Nguyên Tố Có Các Chữ Số Nguyên Tố

Problem

Submissions

Discussions

Viết chương trình đếm xem trong đoạn giữa 2 số a và b có bao nhiêu số là số nguyên tố và tất cả các chữ số của nó cũng là số nguyên tố. Gợi ý : viết hàm 2 là hàm số nguyên tố và hàm kiểm tra tất cả các chữ số của nó là số nguyên tố sau đó duyệt từ a đến b và kết hợp 2 hàm này để kiểm tra.

```
bool ngto(int n){
}

bool csnt(int n){
}

int main(){
    int a, b; cin >> a >> b;
    int dem = 0;
    for(int i = a; i <= b; i++){
        if(csnt(i) && ngto(i)){
            ++dem;
        }
    }
}
```

Input Format

Gồm 2 số nguyên dương a và b.

Constraints

$1 \leq a \leq b \leq 10^7$

Output Format

Ghi ra số lượng số thỏa mãn trên một dòng.

Sample Input 0

```
1234 5678
```

Sample Output 0

```
26
```

[Hàm, Lý Thuyết Số]. Bài 5. Goldbach conjecture

Problem

Submissions

Discussions

Theo Goldbach conjecture, một số nguyên dương chẵn ≥ 4 đều có thể biểu diễn dưới dạng tổng của 2 số nguyên tố. Cho số nguyên dương chẵn $N \geq 4$. Hãy liệt kê các cặp số nguyên tố p, q có tổng đúng bằng N . Ví dụ $N = 6$ ta có 1 cặp số nguyên tố là $3 + 3 = 6$.

Input Format

Dòng đầu tiên đưa vào số lượng bộ test T . Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số chẵn N .

Constraints

$1 \leq T \leq 1000$; $4 \leq N \leq 1000000$

Output Format

Đưa ra kết quả mỗi test theo từng dòng, mỗi dòng là một cặp số thỏa mãn. Chú ý mỗi cặp chỉ liệt kê một lần, không xét đến thứ tự.

Sample Input 0

```
1
10
```

Sample Output 0

```
3 7
5 5
```


[Hàm, Lý Thuyết Số]. Bài 6. Số thuần nguyên tố

Problem

Submissions

Discussions

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 Format

Một dòng hai số nguyên dương tương ứng, cách nhau một khoảng trống.

Constraints

Các số đều không vượt quá 9 chữ số.

Output Format

Viết ra số lượng các số thuần nguyên tố tương ứng

Sample Input 0

```
2345 6789
```

Sample Output 0

```
15
```

[Hàm, Lý Thuyết Số]. Bài 7. Cặp số nguyên tố cùng nhau

Problem

Submissions

Discussions

Hãy liệt kê các cặp số nguyên tố cùng nhau và có giá trị khác nhau trong đoạn $[a,b]$ theo thứ tự từ nhỏ đến lớn.

Input Format

Chỉ có một dòng ghi hai số a, b

Constraints

$1 \leq a \leq b \leq 1000$

Output Format

Các cặp số i,j thỏa mãn được viết lần lượt trên từng dòng theo định dạng (i,j) , theo thứ tự từ điển.

Sample Input 0	(6,31)	(8,29)	(10,39)	(12,41)	(14,43)	(17,28)	(19,39)	(23,24)	(25,46)	(29,43)	(35,36)
	(6,35)	(8,31)	(10,41)	(12,43)	(14,45)	(17,29)	(19,40)	(23,25)	(26,27)	(29,44)	(35,37)
	(6,37)	(8,33)	(10,43)	(13,14)	(15,16)	(17,30)	(19,41)	(23,26)	(26,29)	(29,45)	(35,38)
	(6,41)	(8,35)	(11,12)	(13,15)	(15,17)	(17,31)	(19,42)	(23,27)	(26,31)	(29,46)	(35,39)
	(6,43)	(8,37)	(11,13)	(13,16)	(15,19)	(17,32)	(19,43)	(23,28)	(26,33)	(30,31)	(35,41)
Sample Output 0	(7,8)	(8,39)	(11,14)	(13,17)	(15,22)	(17,33)	(19,44)	(23,29)	(26,35)	(30,37)	(35,43)
	(7,9)	(8,41)	(11,15)	(13,18)	(15,23)	(17,35)	(19,45)	(23,30)	(26,37)	(30,41)	(35,44)
	(7,10)	(8,43)	(11,16)	(13,19)	(15,26)	(17,36)	(19,46)	(23,31)	(26,41)	(30,43)	(35,46)
	(7,11)	(8,45)	(11,17)	(13,20)	(15,28)	(17,37)	(20,21)	(23,32)	(26,43)	(31,32)	(36,37)
	(5,6)	(7,12)	(9,10)	(11,18)	(13,21)	(15,29)	(17,38)	(20,23)	(23,33)	(26,45)	(31,33)
(5,7)	(7,13)	(9,11)	(11,19)	(13,22)	(15,31)	(17,39)	(20,27)	(23,34)	(27,28)	(31,34)	(36,43)
(5,8)	(7,15)	(9,13)	(11,20)	(13,23)	(15,32)	(17,40)	(20,29)	(23,35)	(27,29)	(31,35)	(37,38)
(5,9)	(7,16)	(9,14)	(11,21)	(13,24)	(15,34)	(17,41)	(20,31)	(23,36)	(27,31)	(31,36)	(37,39)
(5,11)	(7,17)	(9,16)	(11,23)	(13,25)	(15,37)	(17,42)	(20,33)	(23,37)	(27,32)	(31,37)	(37,40)
(5,12)	(7,18)	(9,17)	(11,24)	(13,27)	(15,38)	(17,43)	(20,37)	(23,38)	(27,34)	(31,38)	(37,41)
(5,13)	(7,19)	(9,19)	(11,25)	(13,28)	(15,41)	(17,44)	(20,39)	(23,39)	(27,35)	(31,39)	(37,42)
(5,14)	(7,20)	(9,20)	(11,26)	(13,29)	(15,43)	(17,45)	(20,41)	(23,40)	(27,37)	(31,40)	(37,43)
(5,16)	(7,22)	(9,22)	(11,27)	(13,30)	(15,44)	(17,46)	(20,43)	(23,41)	(27,38)	(31,41)	(37,44)
(5,17)	(7,23)	(9,23)	(11,28)	(13,31)	(15,46)	(18,19)	(21,22)	(23,42)	(27,40)	(31,42)	(37,45)
(5,18)	(7,24)	(9,25)	(11,29)	(13,32)	(16,17)	(18,23)	(21,23)	(23,43)	(27,41)	(31,43)	(37,46)
(5,19)	(7,25)	(9,26)	(11,30)	(13,33)	(16,19)	(18,25)	(21,25)	(23,44)	(27,43)	(31,44)	(38,39)
(5,21)	(7,26)	(9,28)	(11,31)	(13,34)	(16,21)	(18,29)	(21,26)	(23,45)	(27,44)	(31,45)	(38,41)
(5,22)	(7,27)	(9,29)	(11,32)	(13,35)	(16,23)	(18,31)	(21,29)	(24,25)	(27,46)	(31,46)	(38,43)
(5,23)	(7,29)	(9,31)	(11,34)	(13,36)	(16,25)	(18,35)	(21,31)	(24,29)	(27,47)	(32,33)	(38,45)
(5,24)	(7,30)	(9,32)	(11,35)	(13,37)	(16,27)	(18,37)	(21,32)	(24,31)	(28,29)	(32,35)	(39,40)
(5,26)	(7,31)	(9,34)	(11,36)	(13,38)	(16,29)	(18,41)	(21,34)	(24,35)	(28,31)	(32,37)	(39,41)
(5,27)	(7,32)	(9,35)	(11,37)	(13,40)	(16,31)	(18,43)	(21,37)	(24,37)	(28,33)	(32,39)	(39,43)
(5,28)	(7,33)	(9,37)	(11,38)	(13,41)	(16,33)	(19,20)	(21,38)	(24,41)	(28,37)	(32,41)	(39,44)
(5,29)	(7,34)	(9,38)	(11,39)	(13,42)	(16,35)	(19,21)	(21,40)	(24,43)	(28,39)	(32,43)	(39,46)
(5,31)	(7,35)	(9,40)	(11,40)	(13,43)	(16,37)	(19,22)	(21,41)	(25,26)	(28,41)	(32,45)	(40,41)
(5,32)	(7,36)	(9,41)	(11,41)	(13,44)	(16,39)	(19,23)	(21,43)	(25,27)	(28,43)	(33,34)	(40,43)
(5,33)	(7,37)	(9,43)	(11,42)	(13,45)	(16,41)	(19,24)	(21,44)	(25,28)	(28,45)	(33,35)	(40,45)
(5,34)	(7,38)	(9,44)	(11,43)	(13,46)	(16,43)	(19,25)	(21,46)	(25,29)	(29,30)	(33,37)	(41,42)
(5,36)	(7,39)	(9,46)	(11,45)	(14,15)	(16,45)	(19,26)	(22,23)	(25,31)	(29,31)	(33,38)	(41,43)
(5,37)	(7,40)	(10,11)	(12,13)	(14,17)	(17,18)	(19,27)	(22,25)	(25,32)	(29,32)	(33,40)	(41,44)
(5,38)	(7,41)	(10,13)	(12,17)	(14,19)	(17,19)	(19,28)	(22,27)	(25,33)	(29,33)	(33,41)	(41,45)
(5,39)	(7,43)	(10,17)	(12,19)	(14,23)	(17,20)	(19,29)	(22,29)	(25,34)	(29,34)	(33,43)	(41,46)
(5,41)	(7,44)	(10,19)	(12,23)	(14,25)	(17,21)	(19,30)	(22,31)	(25,36)	(29,35)	(33,45)	(42,43)
(5,42)	(7,45)	(10,21)	(12,25)	(14,27)	(17,22)	(19,31)	(22,35)	(25,37)	(29,36)	(34,35)	(43,44)
(5,43)	(7,46)	(10,23)	(12,29)	(14,29)	(17,23)	(19,32)	(22,37)	(25,38)	(29,37)	(34,37)	(43,45)
(5,44)	(8,9)	(10,27)	(12,31)	(14,31)	(17,24)	(19,33)	(22,39)	(25,39)	(29,38)	(34,39)	(43,46)
(5,46)	(8,11)	(10,29)	(12,35)	(14,33)	(17,25)	(19,34)	(22,41)	(25,41)	(29,39)	(34,41)	(44,45)
(6,7)	(8,13)	(10,31)	(14,37)	(17,26)	(19,35)	(19,36)	(22,43)	(25,42)	(29,40)	(34,43)	(45,46)
(6,11)	(8,15)	(10,33)	(14,39)	(17,27)	(19,36)	(19,37)	(22,45)	(25,43)	(29,41)	(34,45)	
(6,13)	(8,17)	(10,37)	(14,41)					(25,44)	(29,42)		
(6,17)	(8,19)	(10,39)									
(6,19)	(8,21)	(10,41)									
(6,23)	(8,23)	(10,43)									
(6,25)	(8,25)	(10,45)									
(6,29)	(8,27)	(10,47)									

[Hàm, Lý Thuyết Số]. Bài 8. T-prime

Problem

Submissions

Discussions

Cho số tự nhiên N . Nhiệm vụ của bạn là hãy liệt kê tất cả các số có đúng ba ước số không vượt quá n . Ví dụ $n=100$, ta có các số 4, 9, 25, 49.

Input Format

Số nguyên dương N

Constraints

$1 \leq N \leq 10^{12}$

Output Format

Đưa ra kết quả trên một dòng

Sample Input 0

```
100
```

Sample Output 0

```
4 9 25 49
```

[Hàm, Lý Thuyết Số]. Bài 9. T-prime2

Problem

Submissions

Discussions

Cho số tự nhiên N . Nhiệm vụ của bạn là hãy đếm tất cả các số có đúng ba ước số không vượt quá n . Ví dụ $n=100$, ta có các số 4.

Input Format

Số nguyên dương N

Constraints

$$1 \leq N \leq 10^{12}$$

Output Format

Đưa ra kết quả mỗi theo từng dòng.

Sample Input 0

```
8380000000000
```

Sample Output 0

```
72397
```

[Hàm, Lý Thuyết Số]. Bài 10. Đếm ước của một số nguyên dương

Problem

Submissions

Discussions

Cho phân tích thừa số nguyên tố của một số nguyên dương N , hãy đếm số lượng ước số của số nguyên dương đó. Ví dụ $N = 60 = 2^2 * 3^1 * 5^1$.

Input Format

Dòng đầu tiên là T : số lượng thừa số nguyên tố khác nhau của N T dòng tiếp theo, mỗi dòng là 2 số nguyên p và e lần lượt là thừa số nguyên tố và số mũ tương ứng

Constraints

$1 \leq T \leq 100$; $2 \leq p \leq 100000$; $1 \leq e \leq 100000$;

Output Format

In ra số lượng ước số của N , vì kết quả quá lớn, hãy lấy dư với số 1000000007 ($1e9 + 7$)

Sample Input 0

```
3
2 2
3 1
5 1
```

Sample Output 0

```
12
```


[Hàm, Lý Thuyết Số]. Bài 11. Ước số nguyên tố nhỏ nhất

Problem

Submissions

Discussions

Cho số tự nhiên N . Nhiệm vụ của bạn là in ra ước số nguyên tố nhỏ nhất của các số từ 1 đến N . Ước số nguyên tố nhỏ nhất của 1 là 1. Ước số nguyên tố nhỏ nhất của các số chẵn là 2. Ước số nguyên tố nhỏ nhất của các số nguyên tố là chính nó.

Input Format

Một số N được ghi trên một dòng.

Constraints

$$1 \leq N \leq 100000$$

Output Format

Đưa ra kết quả theo từng dòng

Sample Input 0

```
6
```

Sample Output 0

```
1
2
3
2
5
2
```

[Hàm, Lý Thuyết Số]. Bài 12. Phân tích thừa số nguyên tố

Problem

Submissions

Discussions

Hãy phân tích một số nguyên dương N thành thừa số nguyên tố

Input Format

Số nguyên dương N

Constraints

$$2 \leq N \leq 10^{16}$$

Output Format

Phân tích thừa số nguyên tố của N , xem ví dụ để rõ hơn format.

Sample Input 0

```
60
```

Sample Output 0

```
2^2 * 3^1 * 5^1
```


[Hàm, Lý Thuyết Số]. Bài 13. Bậc của thừa số nguyên tố trong $N!$

Problem

Submissions

Discussions

Cho số tự nhiên N và số nguyên tố P . Nhiệm vụ của bạn là tìm số x lớn nhất để $N!$ chia hết cho p^x . Ví dụ với $N=7, p=3$ thì $x=2$ là số lớn nhất để $7!$ Chia hết cho 3^2 .

Input Format

Cặp số N, p được viết cách nhau một khoảng trống.

Constraints

$1 \leq N \leq 10^{14}; 2 \leq p \leq 5000$

Output Format

Đưa ra kết quả trên một dòng

Sample Input 0

```
7 3
```

Sample Output 0

```
2
```

[Hàm, Lý Thuyết Số]. Bài 14. Đếm chữ số 0 liên tiếp tính từ cuối của N!

Problem

Submissions

Discussions

In ra số lượng chữ số 0 liên tiếp tính từ cuối của N! Ví dụ bạn có $N = 10$, $10! = 3628800$. Như vậy, $10!$ có 2 chữ số 0 liên tiếp tính từ cuối.

Gợi ý : Số 0 ở cuối của N! có được bằng cách nhân số 2 với số 5, cứ mỗi cặp (2, 5) trong N! sẽ tạo ra 1 số 0 ở cuối vì thế số lượng chữ số 0 liên tiếp tính từ cuối của N! sẽ bằng số cặp (2, 5) trong N!, mà số lượng số 2 trong N! bao giờ cũng lớn hơn số lượng số 5! trong N! => Số cặp (2, 5) trong N! chính là số lượng số 5 trong N! hay bậc của 5 trong N!

Input Format

Số nguyên dương N

Constraints

$1 \leq N \leq 10^{18}$

Output Format

In ra số lượng chữ số 0 liên tiếp tính từ cuối của N!. Kết quả lấy dư với 1000000007.

Sample Input 0

10

Sample Output 0

2

[Hàm, Lý Thuyết Số]. Bài 15. Số Sphenic

Problem

Submissions

Discussions

Số nguyên dương N được gọi là số Sphenic nếu N được phân tích duy nhất dưới dạng tích của ba thừa số nguyên tố khác nhau. Ví dụ $N=30$ là số Sphenic vì $30 = 2 \times 3 \times 5$; $N = 60$ không phải số Sphenic vì $60 = 2 \times 2 \times 3 \times 5$. Cho số tự nhiên N , nhiệm vụ của bạn là kiểm tra xem N có phải số Sphenic hay không?

Input Format

Một số nguyên dương N

Constraints

$1 \leq N \leq 10^{18}$

Output Format

Đưa ra 1 hoặc 0 tương ứng với N là số Sphenic hoặc không.

Sample Input 0

```
999923001838986077
```

Sample Output 0

```
1
```

Sample Input 1

```
30
```

Sample Output 1

```
1
```

[Hàm, Lý Thuyết Số]. Bài 16. Số Smith

Problem

Submissions

Discussions

Cho số tự nhiên N . Nhiệm vụ của bạn là hãy kiểm tra N có phải là số Smith hay không. Một số được gọi là số Smith nếu N không phải là số nguyên tố và có tổng các chữ số của N bằng tổng các chữ số của các thừa số nguyên tố trong phân tích của N . Ví dụ $N = 666$ có các thừa số nguyên tố là 2, 3, 3, 37 có tổng các chữ số là 18.

Input Format

Số nguyên dương N

Constraints

$1 \leq N \leq 10^8$.

Output Format

In ra YES nếu N là số Smith, ngược lại in ra NO.

Sample Input 0

```
22
```

Sample Output 0

```
YES
```

```

#include <bits/stdc++.h>
using namespace std;

using ll = long long;

int tong(int n){
    int sum = 0;
    while(n){
        sum += n % 10;
        n /= 10;
    }
    return sum;
}

bool smith(int n){
    int tong1 = tong(n); // tong chu so cua n
    int tong2 = 0;
    int tmp = n;
    for(int i = 2; i <= sqrt(n); i++){
        if(n % i == 0){
            while(n % i == 0){
                tong2 += tong(i);
                n /= i;
            }
        }
    }
    if(tmp == n) return false; // n la snt
    if(n > 1)
        tong2 += tong(n);
    return tong1 == tong2;
}

int main(){
    ll n; cin >> n;
    if(smith(n)) cout << "YES\n";
    else cout << "NO\n";
}

```

[Hàm, Lý Thuyết Số]. Bài 17. Ước số nguyên tố lớn nhất

Problem

Submissions

Discussions

Tìm ước số nguyên tố lớn nhất của một số nguyên dương.

Input Format

Dòng đầu tiên là số lượng test case T; T dòng tiếp theo mỗi dòng là một số nguyên dương N

Constraints

$1 \leq T \leq 500$; $2 \leq N \leq 100000000$

Output Format

Ước số nguyên tố lớn nhất của n in ra mỗi test case trên 1 dòng

Sample Input 0

```
2
10
17
```

Sample Output 0

```
5
17
```

[Hàm, Lý Thuyết Số]. Bài 18. Bình phương nguyên tố 1

Problem

Submissions

Discussions

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 Format

2 số nguyên dương a, b

Constraints

$1 \leq a \leq b \leq 10^6$

Output Format

In ra các số đẹp trong đoạn từ a tới b

Sample Input 0

```
4 50
```

Sample Output 0

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


[Hàm, Lý Thuyết Số]. Bài 19. Bình phương số nguyên tố 2

Problem

Submissions

Discussions

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

2 số nguyên dương a, b

Constraints

$1 \leq a \leq b \leq 10^6$

Output Format

In ra các số đẹp trong đoạn từ a tới b

Sample Input 0

```
3 49
```

Sample Output 0

```
4 8 9 16 25 27 32 36 49
```

[Hàm, Lý Thuyết Số]. Bài 20. Số Chính Phương

Problem

Submissions

Discussions

Kiểm tra một số nguyên có phải là số chính phương hay không? Định nghĩa số chính phương:

https://vi.wikipedia.org/wiki/S%E1%BB%91_ch%C3%ADnh_ph%C6%B0%C6%A1ng

Input Format

Một số nguyên dương N

Constraints

$1 \leq N \leq 10^{18}$

Output Format

In ra YES nếu N là số chính phương, ngược lại in NO

Sample Input 0

169

Sample Output 0

YES

[Hàm, Lý Thuyết Số]. Bài 21. Số Chính Phương trong đoạn

Problem

Submissions

Discussions

In ra các số chính phương trong đoạn từ a tới b. Bài này bạn nào code bằng java thì có thể bỏ qua vì test lớn quá Java không chạy xong trong 8s.

Input Format

2 số nguyên dương a, b

Constraints

$1 \leq a \leq b \leq 10^{12}$

Output Format

In ra các số chính phương trong đoạn giữa 2 số a, b trên một dòng. Các số cách nhau một khoảng trắng.

Sample Input 0

```
10 20
```

Sample Output 0

```
16
```

[Hàm, Lý Thuyết Số]. Bài 22. Số Chính Phương 3

Problem

Submissions

Discussions

Đếm số lượng các số chính phương trong đoạn từ a tới b

Input Format

2 số nguyên dương a, b

Constraints

$1 \leq a \leq b \leq 10^{18}$

Output Format

Số lượng số chính phương trong đoạn $[a, b]$

Sample Input 0

```
1 10000000000
```

Sample Output 0

```
31622
```

[Hàm, Lý Thuyết Số]. Bài 23. Tổng ước số

Problem

Submissions

Discussions

Tính tổng ước của 1 số nguyên dương N .

Input Format

1 số nguyên dương N

Constraints

$1 \leq N \leq 10^{12}$.

Output Format

Tổng ước số của N

Sample Input 0

100

Sample Output 0

217

Sample Input 1

28

Sample Output 1

56

[Hàm, Lý Thuyết Số]. Bài 24. Đếm số lượng ước số của số nguyên 2

Problem

Submissions

Discussions

Đếm số lượng ước của 1 số nguyên dương N

Input Format

Số nguyên dương N

Constraints

$1 \leq N \leq 10^{14}$

Output Format

Số lượng ước của số nguyên dương N.

Sample Input 0

100

Sample Output 0

9

Sample Input 1

28

Sample Output 1

6

[Hàm, Lý Thuyết Số]. Bài 25. Số có số lượng ước là số lẻ

Problem

Submissions

Discussions

Kiểm tra xem một số có số lượng ước số của nó là số lẻ

Input Format

Số nguyên dương N

Constraints

$$1 \leq N \leq 10^{18}$$

Output Format

In ra YES nếu N là số có số ước là số lẻ, ngược lại in NO.

Sample Input 0

```
100
```

Sample Output 0

```
YES
```

Explanation 0

Số 100 có các ước 1, 2, 4, 5, 10, 20, 25, 50, 100. Vậy 100 có 9 ước là số lẻ, nên đáp án là YES.

[Hàm, Lý Thuyết Số]. Bài 26. Số Hoàn Hảo

Problem

Submissions

Discussions

Số hoàn hảo là số có tổng các ước thực sự (Không tính chính nó) bằng chính số đó. Cho một số nguyên dương n , kiểm tra xem n có phải là số hoàn hảo hay không.

Định lý Euclid - Euler : Nếu p là số nguyên tố và $2^p - 1$ cũng là số nguyên tố thì $2^{p-1} * (2^p - 1)$ sẽ tạo thành 1 số hoàn hảo. Ví dụ $p = 2$, $2^2 - 1 = 3$, $2 * 3 = 6 \Rightarrow$ HH Ví dụ $p = 3$, $2^3 - 1 = 7$, $4 * 7 = 28 \Rightarrow$ HH

Tham khảo : <https://blog.28tech.com.vn/c-so-hoan-hao>

Input Format

Số nguyên dương N

Constraints

$1 \leq N \leq 9 * 10^{18}$

Output Format

In YES nếu N là số hoàn hảo, ngược lại in NO

Sample Input 0

28

Sample Output 0

YES

Explanation 0

28 có các ước thực sự là 1, 2, 4, 7, 14 có tổng bằng 28.

[Hàm, Lý Thuyết Số]. Bài 27. Số thuận nghịch có 3 ước nguyên tố

Problem

Submissions

Discussions

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 Format

2 số a, b

Constraints

$1 \leq a \leq b \leq 10^7$

Output Format

In ra các số đẹp trong đoạn a, b . Trong trường hợp không tồn tại số đẹp nào thì in ra -1.

Sample Input 0

```
1 1000
```

Sample Output 0

```
66 222 252 282 414 434 444 474 494 525 555 585 595 606 616 636 646 666 696 777 828 858 868
888 969
```

[Hàm, Lý Thuyết Số]. Bài 28. Số Lộc Phát

Problem

Submissions

Discussions

Một số được gọi là "lộc phát" nếu chỉ có các chữ số 0,6,8. Nhập vào một số nguyên hãy kiểm tra xem đó có phải số lộc phát hay không. Nếu đúng in ra 1, sai in ra 0.

```
bool locphat(long long n){  
      
}  
  
int main(){  
    long long n; cin >> n;  
    if(locphat(n)){  
        cout << 1 << endl;  
    }  
    else{  
        cout << 0 << endl;  
    }  
}
```

Input Format

Số nguyên n

Constraints

$0 \leq n \leq 10^{18}$

Output Format

In ra 1 nếu n là số lộc phát, ngược lại in 0

Sample Input 0

```
60806
```

Sample Output 0

```
1
```

[Hàm, Lý Thuyết Số]. Bài 29. Số thuận nghịch, lộc phát

Problem

Submissions

Discussions

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 các số đẹp trong đoạn giữa 2 số nguyên cho trước, các số cách nhau một dấu cách.

```
bool tn(int n){
}

bool chua6(int n){
    //Tách từng chữ số của n ra, nếu gặp số 6 => return true ngay
    //Sau vòng lặp while => return false
}

bool tong8(int n){
    //Đi tính tổng chữ số của n trước
    //Kiểm tra tong chia 10 dư 8 là return true, ngược lại return false;
}

int main(){
    int a, b; cin >> a >> b;
    for(int i = a; i <= b; i++){
        if(tn(i) && chua6(i) && tong8(i)){
        }
    }
}
```

Input Format

2 số nguyên a, b

Constraints

$1 \leq a \leq b \leq 10^6$

Output Format

Liệt kê các số đẹp trong đoạn, các số viết cách nhau một khoảng trống

Sample Input 0

```
1 400
```

Sample Output 0

```
161
```

[Hàm, Lý Thuyết Số]. Bài 30. Chữ số cuối cùng lớn nhất

Problem

Submissions

Discussions

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

Input Format

Số nguyên dương n

Constraints

$1 \leq n \leq 10^7$

Output Format

Dòng đầu tiên liệt kê các số thỏa mãn, và dòng thứ 2 in ra số lượng số thỏa mãn.

Sample Input 0

```
200
```

Sample Output 0

```
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
```

[Hàm, Lý Thuyết Số]. Bài 31. Phi hàm Euler

Problem

Submissions

Discussions

Đếm số lượng các số nguyên tố cùng nhau với n không vượt quá n .

Input Format

Số nguyên duy nhất n

Constraints

$1 \leq n \leq 10^{12}$

Output Format

Số lượng các số nằm trong đoạn từ 1 tới n mà nguyên tố cùng nhau với n .

Sample Input 0

55

Sample Output 0

40

Sample Input 1

59

Sample Output 1

58

[Hàm, Lý Thuyết Số]. Bài 32. Thừa số nguyên tố thứ K

Problem

Submissions

Discussions

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

Input Format

2 số n,k

Constraints

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

Output Format

In ra thừa số nguyên tố thứ k của n, nếu n không có thừa số nguyên tố thứ k thì in ra -1.

Sample Input 0

28 3

Sample Output 0

7

Sample Input 1

8 5

Sample Output 1

-1

[Hàm, Lý Thuyết Số]. Bài 33. Liệt kê chữ số nguyên tố

Problem

Submissions

Discussions

Liệt kê số lần xuất hiện của chữ số nguyên tố của 1 số theo thứ tự từ nhỏ đến lớn

Input Format

Số nguyên dương n

Constraints

$1 \leq n \leq 10^{18}$

Output Format

Liệt kê các chữ số nguyên tố của n cùng số lần xuất hiện, theo thứ tự từ bé đến lớn. Sau đó cách ra 1 dòng và liệt kê các chữ số nguyên tố của n cùng số lần xuất hiện, nhưng theo thứ tự xuất hiện trong n . Xem thêm test case để rõ hơn.

Sample Input 0

```
2273
```

Sample Output 0

```
2 2
3 1
7 1

2 2
7 1
3 1
```

[Hàm, Lý Thuyết Số]. Bài 34. Số nguyên dương nhỏ nhất chia hết cho x, y, z.

Problem

Submissions

Discussions

Cho 4 số nguyên dương x, y, z, n. Tìm số nguyên dương nhỏ nhất có n chữ số chia hết cho cả x, y, và z.

Input Format

4 số nguyên dương x, y, z, n

Constraints

$(1 \leq x, y, z \leq 10^4); n \leq 16$

Output Format

Kết quả của bài toán, trường hợp không tìm được số thỏa mãn in -1

Sample Input 0

```
2 3 5 4
```

Sample Output 0

```
1020
```

Sample Input 1

```
3 5 7 2
```

Sample Output 1

```
-1
```


[Hàm, Lý Thuyết Số]. Bài 35. Ước chung lớn nhất, bội chung nhỏ nhất

Problem

Submissions

Discussions

Tìm ước chung lớn nhất và bội chung nhỏ nhất của 2 số nguyên

Input Format

2 số nguyên a, b

Constraints

$1 \leq a, b \leq 10^{12}$

Output Format

Ước chung lớn nhất và bội chung nhỏ nhất, dữ liệu đảm bảo BCNN của 2 số không vượt quá số int 64bit

Sample Input 0

```
20 50
```

Sample Output 0

```
10 100
```

[Hàm, Lý Thuyết Số]. Bài 37. Lũy thừa

Problem

Submissions

Discussions

Tính $a^b \% 1000000007$ với a, b nguyên không âm.

Input Format

2 số nguyên không âm a và b

Constraints

$1 \leq a, b \leq 10^{12}$

Output Format

Kết quả của bài toán

Sample Input 0

```
2 10
```

Sample Output 0

```
1024
```

[Hàm, Lý Thuyết Số]. Bài 38. Số lượng ước của N!

Problem

Submissions

Discussions

Đếm số lượng ước của $n!$.

Input Format

Số nguyên dương N

Constraints

$1 \leq n \leq 10^5$

Output Format

Số lượng ước của n giai thừa lấy dư với 1000000007

Sample Input 0

4

Sample Output 0

8

[Hàm, Lý Thuyết Số]. Bài 39. Tính giá trị của hàm F

Problem

Submissions

Discussions

Đối với số nguyên dương n hãy xác định hàm f : $f(n) = -1 + 2 - 3 + \dots + (-1)^n * n$ Nhiệm vụ của bạn là tính $f(n)$ cho một số nguyên n đã cho

Input Format

Số nguyên dương n

Constraints

$1 \leq n \leq 10^{16}$

Output Format

Kết quả của bài toán

Sample Input 0

```
12734
```

Sample Output 0

```
6367
```

[Hàm, Lý Thuyết Số]. Bài 40. Chữ số cuối cùng

Problem

Submissions

Discussions

Cho n , in ra chữ số cuối cùng của 1378^n .

Input Format

Số nguyên không âm n

Constraints

$$0 \leq n \leq 10^{18}$$

Output Format

Chữ số cuối cùng của 1378^n

Sample Input 0

2

Sample Output 0

4

Explanation 0

$$(1378^2) \% 10 = 1898884 \% 10 = 4$$

[Hàm, Lý Thuyết Số]. Bài 41. Số gần thuận nghịch

Problem

Submissions

Discussions

Một số được coi là đẹp nếu chữ số đầu gấp đôi chữ số cuối hoặc ngược lại; đồng thời các chữ số từ vị trí thứ 2 đến gần cuối thỏa mãn là một số thuận nghịch. Ví dụ: các số 36788766; 12345654322 là các số đẹp. Viết chương trình kiểm tra số đẹp theo tiêu chí trên.

Input Format

Số nguyên dương n

Constraints

$$99 \leq n \leq 10^{18}$$

Output Format

Ghi ra YES tương ứng với số đẹp, NO trong trường hợp ngược lại

Sample Input 0

```
122222
```

Sample Output 0

```
YES
```

[Hàm, Lý Thuyết Số]. Bài 42. Hai con mã

Problem

Submissions

Discussions

Đếm số cách đặt 2 con mã trên bàn cờ vua cỡ $k \times k$ với $k = 1, 2, 3, \dots, n$ sao cho chúng không ăn nhau, 2 con mã này được coi là giống nhau.

Input Format

Số nguyên dương n duy nhất

Constraints

$1 \leq n \leq 10000$

Output Format

Gồm n dòng, mỗi dòng là đáp án của bài toán.

Sample Input 0

```
7
```

Sample Output 0

```
0
6
28
96
252
550
1056
```

Explanation 0

Giải thích : với bàn cờ cỡ 1×1 , không có cách nào. Với bàn cờ cỡ 2×2 có 6 cách đặt 2 con mã sao cho chúng không ăn nhau. Tương tự với các cỡ từ 3 tới 7.

[Hàm, Lý Thuyết Số]. Bài 43. Chia táo 1

Problem

Submissions

Discussions

Có n quả táo và m cái hộp, 28Tech muốn tìm số cách chia n quả táo này vào m hộp sao cho hộp nào cũng có ít nhất 1 quả táo. Bạn hãy giúp anh ta nhé.

Input Format

2 số nguyên n, m

Constraints

$1 \leq m \leq n \leq 1000$

Output Format

Kết quả của bài toán lấy dư với 1000000007 ($1e9 + 7$)

Sample Input 0

```
6 3
```

Sample Output 0

```
10
```

Explanation 0

Có 10 cách chia như sau (2, 2, 2) (1, 1, 4) (1, 4, 1) (4, 1, 1) (1, 2, 3) (1, 3, 2) (2, 1, 3) (2, 3, 1) (3, 1, 2) (3, 2, 1)