

# SỞ GIÁO DỤC VÀ ĐÀO TẠO HÀ TĨNH

## HƯỚNG DẪN CHẤM THI

### Bài 1. Cặp số anh em

#### \* Thuật toán đề xuất:

Sử dụng sàng nguyên tố tạo mảng  $p[]$  với ý nghĩa:

- $p[i] = 1$  nếu  $i$  là số nguyên tố và  $p[i] = 0$  trong trường hợp ngược lại.

Duyệt  $x$ :  $1 \rightarrow n - k$

Nếu  $p[x] == 1$  và  $p[x + k] == 1$  thì  $x$  và  $x + k$  là những số nguyên tố khi đó tăng biến kết quả 1 đơn vị.

Độ phức tạp:

- Sàng số nguyên tố  $O(n \log n)$
- Duyệt tìm kết quả  $O(n)$

Độ phức tạp cuối cùng:  $O(n \log n)$

\* Chương trình minh họa bằng ngôn ngữ lập trình C++

```
#include <bits/stdc++.h>
```

```
#define N 1000000
```

```
using namespace std;
```

```
int p[N+3]; int n, k, ans, j;
```

```
void sangngto(int n)
```

```
{
```

```
    for(int i=2; i<=n; i++)
```

```
        p[i]=1;
```

```
    for(int i=2; i<=n; i++)
```

```
        if(p[i]==1)
```

```
            for(j=2*i; j<=n; j+=i)
```

```
                p[j]=0;
```

```
}
```

```
int main()
```

```

{
    freopen("pairpbro.inp","r",stdin);
    freopen("pairpbro.out","w",stdout);
    cin >> n >> k;
    sangngto(n);
    for (int x = 2; x + k <= n; x++)
        if (p[x] == 1 && p[x+k] == 1)
            ans++; cout << ans;
return 0;
}

```

## Bài 2. Đếm ước

### \* Thuật toán đề xuất:

Phân tích T sẽ được phân tích thành tích các thừa số nguyên tố thông qua việc phân tích các giá trị  $A_i$ , thành tích các thừa số nguyên tố.

Giả sử T được phân tích thành tích các thừa số nguyên tố như sau:

$$T = P_1^{k_1} * P_2^{k_2} * \dots * P_m^{k_m}$$

Khi đó kết quả bài toán  $ans = (k_1 + 1) * (k_2 + 1) * \dots * (k_m + 1)$

Độ phức tạp  $O(n * \max(\sqrt{A_i}))$

Tương tự cách làm ở trên.

### \* Chương trình minh họa bằng ngôn ngữ lập trình C++

```

#include <bits/stdc++.h>
#define N 3000000
#define ll long long
#define MOD 123456789
using namespace std;
int p[N+3]; int n;
void phantich (int x)
{
    for (int i=2; i*i<=x; i++)
        {

```

```

        while (x % i == 0)
        { p[i]++; x /= i; }
    }
    if (x > 1) p[x]++;
}

int main()
{
    freopen ("divisor.inp", "r", stdin);
    freopen ("divisor.out", "w", stdout);
    cin >> n;
    for (int x, i=1; i<=n; i++)
    { cin >> x; if (x > 1) phantich(x); }
    ll ans = 1;
    for (int i=2; i<=N; i++)
    ans = (ans*(p[i] + 1))*1LL % MOD;
    cout << ans;
return 0;
}

```

### Bài 3. Số đặc biệt

Trong quá trình nghiên cứu, giáo sư X phát hiện ra một loại số nguyên mới và đặt tên nó là số đặc biệt, một số nguyên dương  $n$  được gọi là số đặc biệt nếu nó thỏa mãn hai tính chất sau:

- 1)  $n$  chia hết cho 3
- 2)  $n$  có đúng 9 ước.

Giáo sư X muốn khảo sát mật độ các số đặc biệt nên nhờ các bạn tham gia thi chọn học sinh giỏi Tin học cấp tỉnh lập trình giải quyết bài toán sau: "Cho hai số nguyên không âm  $a, b$ . Hãy đếm số lượng số đặc biệt trong đoạn  $[a, b]$ "

**Dữ liệu** cho trong tệp văn bản **SDB.INP** gồm:

- Dòng thứ nhất ghi số nguyên dương  $T$  là số bộ dữ liệu.
- $T$  dòng sau, mỗi dòng chứa hai số nguyên dương  $a, b$  ( $a \leq b$ ). Hai

số liên tiếp trên cùng một dòng được ghi cách nhau bởi dấu cách.

Các số ghi trên một dòng được cách nhau bởi dấu cách.

**Kết quả** ghi ra tệp văn bản **SDB.OUT** gồm 7 dòng, mỗi dòng là số lượng số đặc biệt trong đoạn  $[a, b]$  tương ứng với bộ dữ liệu.

Ví dụ:

| SDB. Inp | SDB. Out |
|----------|----------|
| 2        | 0        |
| 1 10     | 1        |
| 220 230  |          |

**Giới hạn:**

- Có 25% test ứng với 25% số điểm thỏa mãn  $a, b \leq 10^3$ ;  $T=1$ ;
- Có 25% test ứng với 25% số điểm thỏa mãn  $a, b \leq 10^4$ ;  $T \leq 100$ ;
- Có 25% test ứng với 25% số điểm thỏa mãn  $a, b \leq 10^6$ ;  $T = 10$ ;
- Có 25% test ứng với 25% số điểm thỏa mãn  $a, b \leq 10^6$ ;  $T \leq 10^5$ .

**\* Đề xuất thuật toán:**

Giá trị  $P$  là số đặc biệt khi:

$P$  chia hết cho 3

$P$  có đúng 9 ước

Từ 2 điều kiện trên suy ra  $P$  có dạng:

Hoặc  $P = 3^8$

Hoặc  $P = x^2 * 3^2$  trong đó  $x$  là một số nguyên tố khác 3.

Xây dựng mảng  $f[]$  với ý nghĩa:  $f[i]$  là số lượng các giá trị nhỏ hơn hoặc bằng  $i$  thỏa mãn chia hết cho 3 và có đúng 9 ước.

Đọc các truy vấn  $(l, r)$ : kết quả của truy vấn là  $f[r] - f[l - 1]$

*\* Chương trình minh họa bằng ngôn ngữ lập trình C++*

```
#include <bits/stdc++.h>
```

```
#define N 1000005
```

```
using namespace std;
```

```
int F[N];
```

```
bool isPrime(int x)
```

```

{
    for (int i=2; i*i<=x; i++)
        if (x % i == 0) return false;
    return true;
}

void init()
{
    F[36] = 1; F[6561] = 1;
    for (int i=5; i<=1000; i++)
        if (isPrime(i) && i*i*9 <= 1000000)
            F[i*i*9] = 1;
    for (int i=1; i<=1000000; i++)
        F[i]+= F[i-1];
}

int main()
{
    init();
    freopen("sdb.inp", "r", stdin);
    freopen("sdb.out", "w", stdout);
    int t;
    cin >> t;
    while (t--)
        { int l,r; cin >> l >> r; cout << F[r] - F[l-1] << '\n'; }
    return 0;
}

```

-----Hết-----