

BEDAO CONTEST

GRAND 06

SPIME

Giả sử ta có một danh sách các số nguyên tố tăng dần và bao quát được giới hạn đề bài. Thì việc của ta chỉ cần đi theo như đề bài. Tổng quát hướng đi: qua phải p đơn vị, đi lên p đơn vị, qua trái $p + 1$ đơn vị, xuống dưới $p + 1$ đơn vị, qua phải $p + 2$ đơn vị, đi lên $p + 2$ đơn vị, qua trái $p + 3$ đơn vị, xuống dưới $p + 3$ đơn vị,... Mỗi bước đi ta duy trì số nguyên tố Agent P thả. Việc cuối cùng ta cần làm là xây dựng danh sách số nguyên tố thỏa mãn.

- Subtask 1: Dùng thuật toán kiểm tra số x có phải là số nguyên tố không với độ phức tạp $O(\sqrt{x})$. Độ phức tạp: $O(\sum_{i=1}^{maxN} \sqrt{i} + N)$.

```
bool isPrime(int n) {
    for (int i = 2; i*i <= n; i++)
        if (n % i == 0) return false;
    return n > 1;
}
```

Hàm $isPrime(x)$ trả về kết quả của mệnh đề “ x là số nguyên tố” trong độ phức tạp $O(\sqrt{x})$

- Subtask 2: Dùng sàng thuật toán nguyên tố để xây dựng danh sách. Độ phức tạp: $O(maxN * \log(maxN) + N)$.

```
void sieve(int N) {
    bool isPrime[N+1];
    for(int i = 0; i <= N; ++i) {
        isPrime[i] = true;
    }
    isPrime[0] = false;
    isPrime[1] = false;
    for(int i = 2; i * i <= N; ++i) {
        if(isPrime[i] == true) {
            // Mark all the multiples of i as composite numbers
            for(int j = i * i; j <= N; j += i)
                isPrime[j] = false;
        }
    }
}
```

BEDAO CONTEST

GRAND 06

Hàm $sieve(maxN)$ giúp ta kiểm tra nhanh số các số trong đoạn $[1; maxN]$ trong độ phức tạp $O(maxN * \log(maxN))$. Mảng $prime[i]$ trả về kết quả của mệnh đề “ i là số nguyên tố”.

Tài liệu có sử dụng hình ảnh (hai đoạn code) từ trang web: <http://vnoi.info/wiki/Home>. Nhân đây mình xin giới thiệu các bạn đến trang web này.