**Bài 1. *(6 điểm)* Đếm số nguyên tố**

**Sub1:**

Với a, b 103. Ta chỉ việc chạy một biến i từ a b, với mỗi biến i kiểm tra chúng có phải là số nguyên tố hay không mất O(i), do đó đpt bài toán được tính trong trường hợp xấu nhất là O((b – a).b).

**Sub2:**

Với a, b 106. Ta cũng chạy một biến i từ a b, với mỗi biến i kiểm tra chúng có phải là số nguyên tố hay không cải tiến thêm là O(), do đó đpt bài toán được tính trong trường hợp xấu nhất là O((b – a).).

**Sub3:**

Với a, b 107. Ở đây ta cần sử dụng kiến thức sàng nguyên tố, ta sẽ dựng sàng nguyên tố với đpt O(107), sau đó sẽ kiểm tra một số có phải là số nguyên tố hay không với đpt O(1). Như vậy đpt bài toán được tính lúc này là O(107 + (b –a)).

**Bài 2: (5 điểm) SỐ MAY MẮN**

Để giải quyết bài toán này ta sử dụng cấu trúc dữ liệu map. Vì ở đây mỗi Key là duy nhất và khác nhau. Tương ứng key là giá trị phần tử mảng đồng thời value là giá trị xuất hiện phần tử đó.

Cụ thể: Ta khai báo một map với tên là mp. Sau đó tương ứng giá trị phần tử a[i] thì mp[a[i]]++. Khi đó a[i] là key và key giống nhau thì giá trị map tăng lên. Mỗi giá trị a[i] ta tìm ra được value. Nói cách khác mỗi key sẽ là 1 value tương ứng. *Giá trị value tương ứng của key đó chính là tần suất xuất hiện của x trong dãy A.*

Chúng ta sử dụng cấu trúc dữ liệu này có thể sử dụng đến 1018 mà vẫn đảm bảo thờ gian chạy 1 giây với giá trị lớn. Vì độ phức tạp O(logn). Đồng thời không cần mảng để lưu, do đó không hạn chế số chỉ số phần tử mảng. Và truy vấn Q tương ứng.

**Bài 3** (*5 điểm).* **DU LỊCH**

**Sub1:**

Ta duyệt lần lượt từng căn phòng trống, mỗi căn phòng trống ta sẽ cho Bình ở đây và cho người bạn dùng những căn phòng gần phòng Bình nhất có thể. Khai báo biến đếm d đến số lượng phòng trống cho cả 2 bên từ vị trí của phòng Bình cho đến khi d=k. Giả sử duyệt đến căn phòng trống thứ X, cho biến L lần lượt sang trái và R lần lượt sang phải cho đến khi chạm giới hạn dãy hoặc d=k. Lúc này khoảng cách từ người bạn xa nhất tới phòng của Bình là:Y= max(X-L, R-X). Kết quả bằng min(kq,Y).

Độ phức tạp: 0(N2).

**Sub2:**

Gọi là số lượng căn phòng trống từ căn phòng đến căn phòng .

Khi duyệt đến một căn phòng trống , ta thực hiện chặt nhị phân tìm khoảng cách nhỏ nhất có thể là . Nếu Bình ở phòng và khoảng cách phòng của các bạn đến phòng của Bình không quá thì phòng của các bạn phải nằm trong phạm vi đế .

Nếu tức là đủ phòng trống cho tất cả người bạn và Bình trong phạm vi đó. Lúc này ta sẽ cập nhập kết quả .

Độ phức tạp:

**Bài 4:** *(4 điểm)* **ATM**

Có thể giải bài toán với cách giải tham lam

Muốn số tờ tiền càng ít càng tốt thì mệnh giá cao nhất sẽ được rút trước và rút được mệnh giá cao nhất càng nhiều càng tốt.

Tuy nhiên ý tưởng tham lam chỉ đúng trong một vài trường hợp không phải bài nào cũng dùng ý tưởng tham lam. Đây là bài toán điển hình cho ý tưởng tham lam bị sai.

Phương pháp trong bài toán này sử dụng quy hoạch động.

Bài toán phân ra như sau: Số tờ tiền ít nhất nhặt ra trong n loại d1, d2, …dn. Có tổng bằng S. Chọn tập có tổng bằng S và số tờ tiền ít nhất.

Quay về bài toán quen thuộc có tổng bằng S

Gọi F[j]= số tờ tiền ít nhất để để có tổng bằng j. Suy ra F[S] là kết quả của bài toán.

F[0]=0. Xét tờ tiền có mệnh giá d[i] (1<=d[i]<=n)

Xét tiền cần rút là j ( j thuộc đoạn d[i] đến S) có hai khả năng xảy ra. j có thể chạy từ d[i] đến S. Hoặc j chạy từ S đến d[i]

F[j]=min(F[j], F(j-d[i]+1).

d[i] tham gia vào tổng j .

Vậy j chạy từ d[i] đến S hay j chạy ngược từ S đến d[i].

Bài toán này J chạy xuôi từ d[i] đến S vì d[i] được dùng nhiều lần.