**Báo Cáo Tuần 2 - Thực Hành DSA**

1. **LinearSearch:**

Duyệt tuần tự các phần tử trong mảng tử 0 -> n rồi kiểm tra nếu tồn tại giá trị thỏa mãn thì trả về index i thỏa a[i]==k. Nếu không tồn tại trả về -1

1. **SentinelSearch:**

Tạo biến last để trữ giá trị cuối cùng của mảng, gán giá trị cuối cùng = k ( lính canh, để chặn lại vòng lặp bên dưới )

Duyệt tuần tự các phần tử trong mảng với biến đếm i, nếu i chưa đạt đến giá trị cuối cùng hoặc giá trị cuối cùng bằng biến last ban đầu thì ta trả về vị trí i. Ngược lại, các trường hợp khác có nghĩa là không tồn tại giá trị cần tìm trong mảng.

1. **Tìm giá trị nhỏ nhất trong hàm sort chuyển vị**

Xét tính chất “số nhỏ nhất” ta thấy min là giá trị chặn của dãy số, hay nói cách khác nó làm đứt mạch liên tục của chuỗi theo chiều xuôi. Dựa vào tính chất đó, ta sử dụng phương pháp binary search để duyệt, bởi vì giá trị cần tìm là “đặc biệt trong mảng có quy luật”. Đoạn code trên em giả sử hàm cần tìm sắp xếp theo thứ tự tăng dần. Và điều kiện đặc biệt ở đây là kẹp giữa hay ***left > min > right***

Cùng với đó là kiểm tra trường hợp đặc biệt nếu phần tử đầu bé hơn cuối rồi ( mảng tăng dần chưa đảo vị trí ) thì trả về phần tử đầu luôn.

Thêm bước return 0 ngoài vòng lặp để tránh lỗi và cảnh báo vì khai báo hàm int.

1. **C**
2. **D**
3. **Tìm 2 số trong mảng sao cho tổng thỏa Target cho trước**

* **Cách 1: *Bruteforce***:

Duyệt tuần tử cả mảng với 2 vòng for lồng nhau. Điều kiện trả ra hàm là **a[i]+a[j] == target** với j chạy từ i + 1 tới n và i chạy từ 0 tới n.

* **Cách 2 : *BinarySearch*:** Chạy 2 vòng lặp lồng nhau