

BÁO CÁO THỰC HÀNH DSA TUẦN 02
24120261 – Đặng Bùi Thế Bảo

1. Exercise 1 - Linear Search

Mô tả: Tìm vị trí đầu tiên của số K trong mảng bằng thuật toán Linear Search.

Cách làm:

- Bắt đầu từ chỉ số 0 của mảng.
- Dùng một vòng lặp để duyệt qua từng phần tử trong mảng.
- Tại mỗi phần tử, so sánh giá trị của nó với số K cần tìm.
- Nếu tìm thấy phần tử bằng K, trả về chỉ số của phần tử đó ngay lập tức.
- Nếu duyệt hết mảng mà không tìm thấy, trả về -1 để báo hiệu K không tồn tại trong mảng.

2. Exercise 2 - Linear Search with Sentinel

Mô tả: Tìm K trong mảng bằng Linear Search với Sentinel.

Cách làm:

- Lưu giá trị phần tử cuối cùng của mảng vào một biến tạm (last).
- Thay phần tử cuối cùng bằng K để làm "sentinel".
- Dùng vòng lặp while để duyệt mảng từ đầu, tăng chỉ số i cho đến khi gặp phần tử bằng K.
- Sau khi thoát vòng lặp, khôi phục giá trị cuối cùng của mảng bằng giá trị ban đầu (last).
- Kiểm tra xem chỉ số i tìm được có nhỏ hơn vị trí cuối cùng không, hoặc phần tử cuối ban đầu có phải là K không; nếu đúng, trả về i, ngược lại trả về -1.

3. Exercise 3 - Tìm phần tử nhỏ nhất trong mảng xoay

Mô tả: Tìm phần tử nhỏ nhất trong mảng đã sắp xếp tăng dần nhưng bị xoay.

Cách làm:

- Sử dụng Binary Search với hai con trỏ right và left, lần lượt là đầu và cuối mảng.
- Trong mỗi bước, tính chỉ số giữa (mid) của khoảng right và left: $mid = left + (right - left) / 2$.
- Đặt điều kiện (left = right) => trả về giá trị tại left.
- So sánh giá trị tại mid với giá trị tại right:
 - Nếu $arr[mid] < arr[right]$, phần nhỏ nhất nằm ở nửa bên trái => gọi đệ quy về phía bên trái, tức là từ left \rightarrow mid
 - Ngược lại, gọi đệ quy về phía bên phải tức là mid+1 \rightarrow right

4. Exercise 4 - Tối thiểu dung lượng tàu

Mô tả: Tìm dung lượng tối thiểu của tàu để vận chuyển hết gói hàng trong số ngày cho trước.

Cách làm:

- Xác định dung lượng tối thiểu(low): là giá trị max của mảng
Dung lượng tối đa(high): là tổng của mảng
- Dùng Binary Search trên khoảng này:
 - Tính dung lượng giữa (mid) của khoảng.
 - Với mỗi mid, thử chia mảng weights thành các nhóm sao cho tổng mỗi nhóm không vượt quá mid, đếm số ngày cần thiết.
 - Nếu số ngày vượt quá giới hạn, tăng dung lượng bằng cách gán $low = mid + 1$.
 - Nếu số ngày nhỏ hơn hoặc bằng giới hạn, giảm dung lượng bằng cách gán $high = mid - 1$.
- Khi Binary Search kết thúc, $high + 1$ là dung lượng tối thiểu cần tìm.

5. Exercise 5 - Độ dài nhỏ nhất của mảng con

Mô tả: Tìm độ dài nhỏ nhất của mảng con có tổng lớn hơn hoặc bằng target.

Cách làm:

- Khởi tạo hai con trỏ j bằng giá trị low , $k = low + 1$ để xác định cửa sổ, cùng với biến sum để tính tổng cửa sổ.
- Di chuyển con trỏ k để thêm phần tử vào cửa sổ, tăng sum.
- Khi sum lớn hơn hoặc bằng target, ghi lại độ dài của cửa sổ hiện tại ($end - start + 1$), rồi thu hẹp cửa sổ bằng cách tăng start và giảm sum.
- Lặp lại quá trình này cho đến khi duyệt hết mảng.
- Trả về độ dài nhỏ nhất tìm được; nếu không có cửa sổ nào thỏa mãn, trả về 0.

6. Exercise 6 - Kiểm tra tổng hai số

Mô tả: Kiểm tra xem có hai số trong mảng sắp xếp có tổng bằng target không.

Cách làm:

- Đặt hai con trỏ: left ở đầu mảng (chỉ số 0) và right ở cuối mảng (chỉ số $n-1$).
- Tính tổng của $arr[left]$ và $arr[right]$:
 - Nếu tổng bằng target, trả về "YES".
 - Nếu tổng nhỏ hơn target, tăng left để tăng tổng.

- Nếu tổng lớn hơn target, giảm right để giảm tổng.
- Lặp lại cho đến khi left vượt qua right; nếu không tìm thấy, trả về "NO".

7. Exercise 7 - Tìm bộ ba số có tổng bằng 0

Mô tả: Tìm tất cả bộ ba số trong mảng có tổng bằng 0, không trùng lặp.

Cách làm:

- Sắp xếp mảng theo thứ tự tăng dần để dễ xử lý.
- Duyệt qua từng phần tử $arr[i]$ làm phần tử đầu tiên của bộ ba.
- Với mỗi $arr[i]$, dùng hai con trỏ left ($i+1$) và right ($n-1$) để tìm hai số còn lại sao cho $arr[i] + arr[left] + arr[right] = 0$.
- Nếu tổng bằng 0, lưu bộ ba vào kết quả, tăng left và giảm right, đồng thời bỏ qua các giá trị trùng để tránh lặp.
- Nếu tổng nhỏ hơn 0, tăng left; nếu lớn hơn 0, giảm right.
- Lặp lại cho đến khi duyệt hết mảng, trả về danh sách các bộ ba.