

BÁO CÁO THỰC HÀNH DSA TUẦN 5:

24120261 – Đặng Bùi Thế Bảo

***STACK:** Ngăn xếp là cấu trúc dữ liệu theo nguyên lý **LIFO (Last In, First Out)**, nghĩa là phần tử được thêm vào sau cùng sẽ được lấy ra trước tiên. Các thao tác chính của ngăn xếp bao gồm:

- **Push:** Thêm phần tử vào đỉnh ngăn xếp.
- **Pop:** Lấy phần tử ra khỏi đỉnh ngăn xếp.
- **Size:** Kiểm tra số lượng phần tử trong ngăn xếp.
- **IsEmpty:** Kiểm tra xem ngăn xếp có rỗng hay không.
- **PrintStack:** In ra tất cả các phần tử trong ngăn xếp.
- **Cách tiếp cận cho ngăn xếp (Stack)**
 - **Khởi tạo Ngăn xếp:** Chúng ta sử dụng một cấu trúc NODE để đại diện cho các phần tử trong ngăn xếp, mỗi phần tử chứa một khóa và một con trỏ p_next trỏ đến phần tử kế tiếp. Trong cấu trúc Stack, chúng ta chỉ cần một con trỏ top trỏ đến phần tử đầu tiên (đỉnh ngăn xếp).
 - **Thao tác Push:** Khi thêm phần tử, con trỏ của phần tử mới được trỏ tới phần tử hiện tại ở đỉnh ngăn xếp và sau đó top được cập nhật để trỏ đến phần tử mới.
 - **Thao tác Pop:** Khi lấy phần tử ra, chúng ta lấy giá trị từ phần tử đầu tiên (đỉnh ngăn xếp) và cập nhật con trỏ top trỏ tới phần tử tiếp theo.
 - **Kiểm tra kích thước và trạng thái rỗng:** Để kiểm tra kích thước, chúng ta duyệt qua danh sách liên kết. Để kiểm tra trạng thái rỗng, chỉ cần kiểm tra con trỏ top có phải là nullptr hay không.
 - **In ra trạng thái ngăn xếp:** Duyệt qua tất cả các phần tử từ top xuống dưới và in ra giá trị của từng phần tử vào file output.txt.

***QUEUE:** Hàng đợi là cấu trúc dữ liệu theo nguyên lý **FIFO (First In, First Out)**, nghĩa là phần tử được thêm vào trước sẽ được lấy ra trước. Các thao tác chính của hàng đợi bao gồm:

- **Enqueue:** Thêm phần tử vào cuối hàng đợi.

- **Dequeue:** Lấy phần tử ra khỏi đầu hàng đợi.
- **Size:** Kiểm tra số lượng phần tử trong hàng đợi.
- **IsEmpty:** Kiểm tra xem hàng đợi có rỗng hay không.
- **PrintQueue:** In ra tất cả các phần tử trong hàng đợi.
- **Cách tiếp cận cho hàng đợi (Queue)**
 - **Khởi tạo Hàng đợi:** Hàng đợi được đại diện bởi một cấu trúc NODE, tương tự như ngăn xếp. Tuy nhiên, trong hàng đợi, chúng ta cần hai con trỏ: p_head để chỉ vào phần tử đầu tiên và p_tail để chỉ vào phần tử cuối cùng.
 - **Thao tác Enqueue:** Khi thêm phần tử, chúng ta tạo một phần tử mới, nối nó vào cuối danh sách, và cập nhật con trỏ p_tail trỏ tới phần tử mới.
 - **Thao tác Dequeue:** Khi lấy phần tử ra, chúng ta lấy giá trị của phần tử đầu tiên (tại p_head), cập nhật p_head để trỏ tới phần tử tiếp theo trong danh sách.
 - **Kiểm tra kích thước và trạng thái rỗng:** Kiểm tra trạng thái rỗng cũng tương tự như ngăn xếp, bằng cách kiểm tra con trỏ p_head. Kích thước có thể được tính bằng cách duyệt qua tất cả các phần tử trong hàng đợi.
 - **In ra trạng thái hàng đợi:** Duyệt qua tất cả các phần tử từ p_head đến p_tail và in ra giá trị của từng phần tử, với dấu mũi tên "->" để thể hiện mối quan hệ giữa các phần tử.