Stack (스택)

- LIFO (Last In First Out)
- 시간복잡도 : O(1) ↳ 맨끝에서만 연산이 일어나므로
- 연산 : push & pop, top, empty, size
- empty일때 top=-1, 자료가 1개일때 top=0, 자료가 2개일 때 top=1 …

  ↳ top : 배열의 index 처럼 생각!
- std::stack ↳ #include <stack>

  ↳ stack<자료형> 이름;

  ↳ 함수사용시 '이름.pop()' 과 같이 사용하면 된다.

Queue (큐)

- FIFO (First In First Out)
- 시간복잡도 : O(1) ↳ 왼쪽 끝에서 삭제, 오른쪽 끝에서 삽입 (끝에서 연산) 하므로
- 연산 : push & pop, front & back (추적), empty, size
- 배열로 원형큐를 구현한다고 했을 때, 배열 0번은 비워둔다.

  front는 맨 처음 데이터 위치의 바로 앞을 가리키며, rear은 맨 마지막 데이터 위치를 가리킨다.

  ∴ front와 rear이 같은 곳에 위치되어 있으면 empty 이고, rear+1 이 front일 때 (즉 둘이 1칸만 차이날 때) full이다.
- std::queue ↳ #include <queue>

  ↳ queue<자료형> 이름;

  ↳ 함수 사용시 '이름.front()' 와 같이 사용하면 된다.

Deque (덱)

- Stack + Queue : 자료의 양끝에서 연산이 이루어짐
- 시간복잡도 : O(1) ↳ 양 끝에서 연산이 이루어지므로
- 연산 : push_front, push_back & pop_front, pop_back , front , back ,
  empty, size
- std:: deque ↳ #include <deque>
  ↳ deque<자료형> 이름;
  ↳ 함수 사용시 '이름.front()' 와 같이 사용하면 된다.