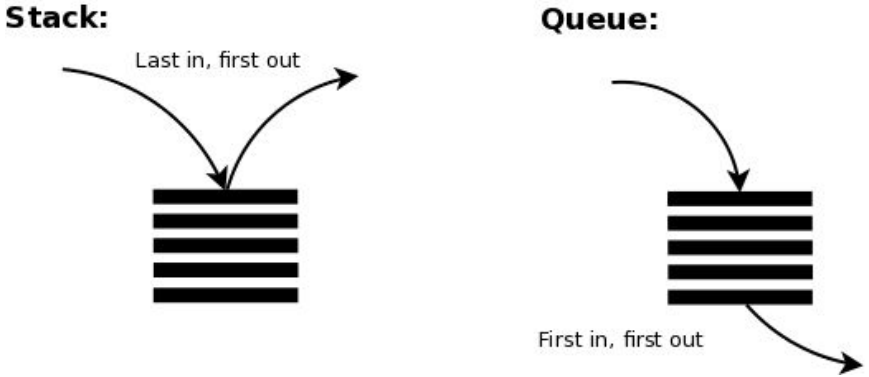
Stacks & Queues

Data Structures
Fall 2023

Stacks

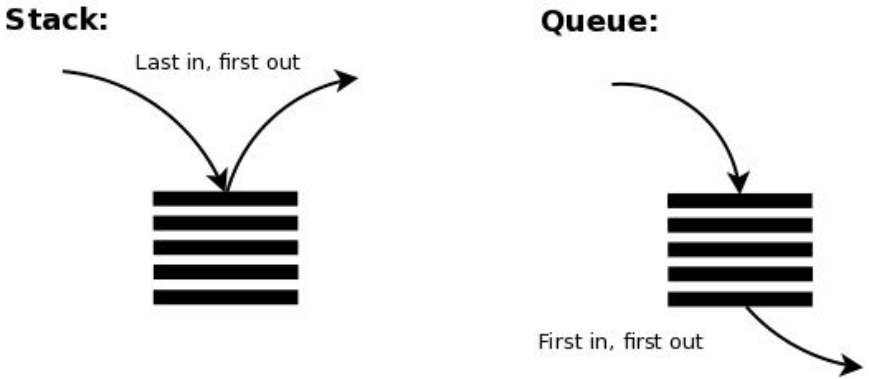


- LIFO
  - Last in First Out
- Push
  - Adds an element to the **top** of the stack
- Pop
  - Returns the **top** element **and** removes it.
- Peek / Top
  - Returns the top element but does not remove it.
- isEmpty
  - Checks whether the stack is empty
- Size
  - Gives the size of the stack

Complexities (Stack)

- Push: O(1)
- Pop: O(1)
- Peek: O(1)
- Search: O(n)

Queue



- FIFO
  - First in First Out
- Enqueue
  - Adds an item to the **end** of the queue
- Dequeue
  - Removes the item from the **front** of the queue
- Peek / Front
  - Gives the top element but does not remove it.
- IsEmpty
  - Returns boolean value to see if the DS is empty or not
- Size
  - Gives the size of the queue

Complexities

- Enqueue: O(1)
- Dequeue: O(1)
- Peek: O(1)
- Search: O(n)

Caution

- Underflow and overflow
  - If the stack/queue is empty and you are trying to remove something
  - If the stack/queue is full and you are trying to add something
- Implementation
  - If you want them to be fully dynamic, you need to use a dynamic data structure
  - If they are bounded/limited, you can use a static data structure (array)

Used for

- Stacks
  - Expression evaluation
  - Backtracking algorithms
  - Syntax parsing
  - Maintaining function calls (call stack)
- Queues
  - Breadth-first searching algorithms
  - Scheduling processes
  - Buffering data streams

Stacks use cases

- Function calls
  - When a function is called, its environment and return address are pushed onto a call stack, and when the function execution is complete, they are popped off.
- Expression evaluation
  - Evaluate prefix, infix and postfix expressions.
  - Operators and operands are pushed onto a stack in a manner that allows for the evaluation of complex expressions.
- Page History
  - In web browsers, stacks can be used to keep track of the previous web pages
  - Undo

...

Queue use cases

- All systems which use *first come first serve*
- Order management
- Print queue
- Asynchronous data transfer
  - Queues are used to manage in data communication to make sure the packets are received and sent in the right order.
- Call center
- etc.

Implementation

- We are going to implement them by using LinkedLists.
- Implement the linked list
- Create a stack class and add methods