Stacks

Overview

- A one-ended linear data structure that models an actual stack (push and pop operations)
- LIFO

Why Use It?

- Used by undo mecahnisms
- Used in compilers to check for matching brackets and braces in the right order
- Used to support recursion by keeping track of previous function calls
- Used for Depth First Search (DFS) on a graph

Big O Analysis

Operation	Big O Notation	Explanation
Pushing	O(1)	A reference to the top of the stack allows for constant time
Popping	O(1)	A reference to the top of the stack allows for constant time
Searching	O(n)	An element needed isn't always at the top of the stack
Peeking	O(1)	A reference to the top of the stack allows for constant time
Size	O(n)	

Code Implementation

- Stacks can be implemented as arrays, singly linked list, or doubly linked list
- https://www.geeksforgeeks.org/stack-data-structure-introduction-program/

Techniques / Problems

- Brackets problem

```
// Let S be the stack

for bracket in string:
  revBracket = getReversedBracket(bracket)

  if(isLeftBracket(bracket)):
    S.push(bracket)
  elif(S.pop() != revBracket or S.isEmpty()):
    return false

  return S.isEmpty()
```

- Tower of Hanoi