



4.2.3 Stacks

Stack

Definition

A stack is a data structure that follows the LIFO principle (Last In, First Out).

Operations

Push adds an item to the top of the stack.

- Check if the stack is full
- If not, insert an item at the top of the stack
- If it is full, raise an error message

Pop removes and returns the item from the top of the stack.

- Check if the stack is empty
- If not, return the item from the top of the stack
- If it is empty, raise an error message.

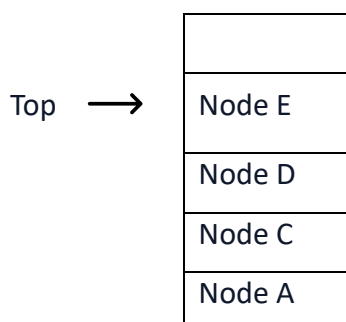
Peek returns the item from the top of the stack without removing it.

Implementation

One way of implementing a stack is:

- Using a 1D array.
- Creating a pointer.

The **top pointer** stores the position of the item at the top of the stack.



When the stack is empty, the top pointer is usually set to -1. This shows that there are no items currently stored in the stack.

Trying to push an item onto a full stack is called an **overflow error**. Trying to pop an item from an empty stack is called an **underflow error**.





Example Code

```
Push(item)

    if Top == array.length - 1:
        print("Stack is full")
    else:
        Top = Top + 1
        array[Top] = item
```

```
Pop()

    if Top == -1:
        print("Stack is empty")
        return -1
    else:
        item = array[Top]
        Top = Top - 1
        return item
```

