DSA.ai

Type the Question

Enter your question here:

Stack Implementation

Run LLM API call took 220.09847569465637 seconds.

Stack Implementation

In computer science, a stack is a data structure that follows the Last In First Out (LIFO) principle, meaning that the last element added to the stack will be the first one to be removed. A stack can be implemented using an array or a linked list, and it is commonly used in many applications such as parsing expressions, undo/redo functionality, and more.

Classification: Arrays

In this problem, we will focus on implementing a stack using an array.

Complete Python Code

```python
class Stack:
    def __init__(self):
        self.stack = []

    def push(self, element):
        self.stack.append(element)
```

```python
    def pop(self):
        if len(self.stack) == 0:
            return None
        else:
            return self.stack.pop()

    def peek(self):
        if len(self.stack) == 0:
            return None
        else:
            return self.stack[-1]

    def size(self):
        return len(self.stack)

    def is_empty(self):
        return len(self.stack
```

```
) == 0
```

Sample Usage
```python
# Create a new stack
stack = Stack()

# Add elements to the stack
stack.push(1)
stack.push(2)
stack.push(3)

# Peek at the top element of the stack
print(stack.peek()) # Output: 3

# Pop the top element from the stack
print(stack.pop()) # Output: 3

# Check if the stack is empty
if stack.is_empty():
    print("The stack is empty.")

# Add more elements to the stack
stack.push(4)
stack.push(5)
stack.push(6)
```

```
# Peek at the top element of the stack
print(stack.peek()) # Output: 6
```

Summary

A stack is a simple data structure that can be implemented using an array or a linked list. The push() method adds an element to the top of the stack, while the pop() method removes and returns the top element from the stack. The peek() method returns the top element without removing it. The size() method returns the number of elements in the stack. The is_empty() method checks if the stack is empty or not.

Hints

- Use the `append()` method to add an element

to the end of the array.

- Use the `pop()` method to remove and return the last element from the array.
- Use the `len()` function to get the length of the array.
- Use the `-1` index to access the last element of the array.

Similar Problems

- Implement a queue using an array.
- Implement a stack using a linked list.

Response received!