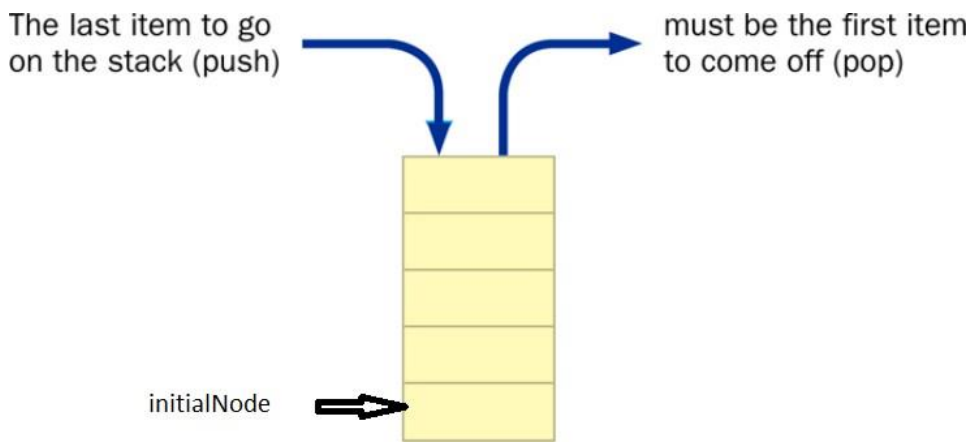
SENG102

Lab Assignment 11

Question 1



In this question, you are asked to implement the stack data structure using linked lists. A stack has two main operations which are push and pop. The push operation inserts an item at the top of the stack. The pop operation removes and retrieves the item at top of the stack. Node.java, Stack.java and DriveStack.java files are given in the LMS. You are asked to implement push and pop methods in the Stack.java file. The expected output is given below.

Note: Pop should return -1 when the stack is empty.

Expected Output:

```
Stack contains elements: 15 12 100 50
50
100
Stack contains elements: 15 12
12
15
-1
```

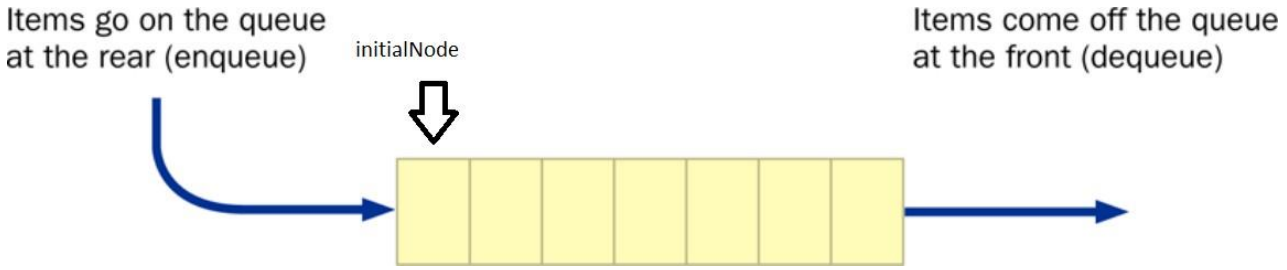
Push Algorithm:

- If the stack is empty:
  - initialNode = newNode
- If the stack is not empty:
  - Traverse the stack until the last node
  - lastNode.next = nodeToAdd

Pop Algorithm:

- If the stack is empty:
  - Return -1
- If the stack contains only one value:
  - Return its value and set the initialNode to null.
- If the stack contains multiple nodes:
  - Traverse until the last node. Make sure the keep the node that precedes the current node in a variable.
  - preecedingNode.next = null
  - Return the value of the current node.

Question 2



In this question, you are asked to implement queue data structure using linked lists. In contrast to stacks, queues pushes new items to the bottom (where initialNode is located). However, it still pops from the top.

- You have implemented stacks in the question 1. You are expected to change the push method so that it pushes to the top of the queue.
- You are given **NodeString.java** file which contains a node structure that keep a string value instead of integer value. Make sure that your queue methods work with NodeString type.
- You are given **DriveQueue.java** file to test your code. Make sure it produces the following output.

Expected Output:

```
Queue contains elements: CMPE233 CMPE114 CMPE113
CMPE113
CMPE114
```

Push Algorithm:

- The initialNode will be assigned to the new node's next attribute.
- The new node will be made the new initialNode.

Submission: Please submit Stack.java and Queue.java files.