

## Lab 7

### boolean empty() algorithm:

1. If size of theData is zero then stack is empty, return true
2. Else, return false

### E peek() algorithm:

1. If stack is empty, throw an exception that the stack is empty
2. Else, return the element of theData at size - 1

### E pop() algorithm:

1. If stack is empty, throw an empty state exception
2. Use remove method of theData to remove the element at size - 1 and store in a temporary variable
3. Return the the temporary variable that contains the removed element

### E push(E obj) algorithm:

1. Use add method of theData to add obj to the top
2. Return obj

### Question 1

- a. The first line adds "Jane" to the top of the stack
- b. The second line adds "Joseph" to the top of the stack
- c. The third line stores "Joseph" in the variable top and removes "Joseph" from the stack
- d. The fourth line stores "Jane" in the variables nextTop without removing it from the stack

Stack a				
Jane	Joseph	top = "Joseph"	nextTop = "Jane"	
Jonathan	Jane	Jane	Jane	
Dustin	Jonathan	Jonathan	Jonathan	
Robin	Dustin	Dustin	Dustin	
Debbie	Robin	Robin	Robin	
Rich	Debbie	Debbie	Debbie	
	Rich	Rich	Rich	

Stack b				
Jane	Joseph	top = "Joseph"	nextTop = "Jane"	
Dustin	Jane	Jane	Jane	
Robin	Dustin	Dustin	Dustin	
Debbie	Robin	Robin	Robin	
Rich	Debbie	Debbie	Debbie	
	Rich	Rich	Rich	

Stack c				
Jane	Joseph	top = "Joseph"	nextTop = "Jane"	
Philip	Jane	Jane	Jane	
Dustin	Philip	Philip	Philip	
Robin	Dustin	Dustin	Dustin	
Debbie	Robin	Robin	Robin	
Rich	Debbie	Debbie	Debbie	
	Rich	Rich	Rich	

**Question 2**

While the stack is not empty, print the name at the top and remove it from the stack.

**Question 3**

Peek would print the name at the top of the stack but would not remove it. As a result, the stack would never be empty and we would have an infinite loop that would keep printing "Philip".