In this quiz, we will develop various methods associated with a java class MyIntStack which encodes a stack of integers. This class has the following instance-variable:

```
private int[] data;
```

The number of items in the stack is exactly equal to the length of this array, and the top of the stack (where items are inserted and deleted) is data[0].

a) (2 marks) Write a no-parameter Java method size that returns the number of items in the invoking stack-object.

```
public int size(){
    return(data.length);
}
```

b) (3 marks) Write a no-parameter Java method isEmpty that returns true if there are no items in the stack and false otherwise.

```
public boolean isEmpty(){
    return(data.length == 0? true: false);
}
```

c) (5 marks) Write a one-parameter Java method push(newVal) that adds integer-value newVal to the top of the stack.

```
public void push(int newVal){
    int[] tempData;

    tempData = new int[data.length + 1];
    System.arraycopy(data, 0, tempData, 1, data.length);
    tempData[0] = newVal;
    data = tempData;
}
```
d) (5 marks) Write a no-parameter Java method peek that returns the integer value at the top of the stack. If the stack is empty when this method is called, an EmptyStackException exception is thrown.

```java
public int peek(){
    int[] tempData;
    int temp;

    if (isEmpty())
        throw new EmptyStackException();
    return(data[0]);
}
```

e) (5 marks) Write a no-parameter Java method pop that returns the integer value at the top of the stack and removes this value from the top of the stack. If the stack is empty when this method is called, an EmptyStackException exception is thrown.

```java
public int pop(){
    int[] tempData;
    int temp;

    if (isEmpty())
        throw new EmptyStackException();
    temp = data[0];
    tempData = new int[data.length - 1];
    System.arraycopy(data, 1, tempData, 0, tempData.length);
    data = tempData;
    return(temp);
}
```