LAB #6 – Linear Data Structures

You will implement a simplified stack and queue. Yes you’ve already done them. But this time (in a step backwards) you’ll implement them using arrays rather than linked structures.

1. Stack-like behavior. A stack is a first in last out (FILO) structure. So the top of the stack will be the last item entered. You are not required to implement any other features of the formal Stack data structure. Be careful when you research this. You could make your program more difficult.

You will create a node. You will need to add an element and to take off an element. You can use a pointer to the top element or keep track of the array index. You should have these functions for your Stack:

```c
void push(char Value)
void pop()
char peek()
bool isEmpty()
```

This is all that is required.

2. Queue-like behavior. A queue is a first in first out (FIFO) structure. You will need a pointer to the front and to the back of the queue.

You will create a node. You can use a front pointer and a rear pointer. Or you can track the array index of the first and last elements. You will only add elements at the back and only take off from the front. You should have these functions for your Queue:

```c
void add(char Value)
char remove()
char peek()
bool isEmpty()
```

In this case Node will just contain a char value. But use an array of Node to make things easier in the future.

An array, eh? Should it be dynamic or static? How will tell if it’s full? Does it matter? Think about this before you start coding. It’s very easy to code yourself into a corner on this one. ☺️