Homework 3 – STACKS

All the starter files for this homework are in the CS5008_ClassRepo repository in Github. Please pull them as necessary.

In this assignment, you will extend the STACK abstract data type by implementing using a linked list. This work will use the api created in class where STACK was implemented using VECTOR ADT. The start code for the function definitions and the header file and the main driver has been given. The first part is for you to look through the code and understand what is being done before embarking on your implementation.

There are four parts for you to implement:

1) The push function (10 points) – please see the code. The function receives the address of the opaque pointer and some data item. The post condition is that the function will return a FAILURE or SUCCESS; it will add new nodes to the stack and the new node should store the value.
2) The pop function (5 points) – will receive the address of the opaque object. It will then remove the node which is on top of the stack.
3) The peek function (5 points) – will return value stored by the node on top of the stack
4) The deallocate function (10 points) – will free all memory attached to the stack. Note that there is memory of stack and memory of the nodes.

Submit all your code in your github repo. Make sure that prior to pushing to github you have tested the code in the Khoury server.