LAB #8 – Writing a C Program: A Stack

In order to get credit for the lab, you need to be checked off by the end of lab. You can earn a maximum of 3 points for lab work completed outside of lab time, but you must finish the lab before the next lab. For extenuating circumstances, contact your lab TAs and Jennifer Parham-Mocello.

Reminder: All of our labs involve paired programming. You do not have to keep the same partner for each lab, but you MUST work with someone in each lab!!! First, find a partner for this lab. It can be the same partner from the previous lab or a different partner.

(7 pts) Implement a Stack in C
In this lab, you are going to implement a stack data structure using a dynamic array in C. A stack is a LIFO (last in, first out) structure, where you only push and pop items from the top. For example, a deck of cards, a stack of plates, etc. We will implement the stack using a dynamic array and a struct holding the array and a top, which stores the location of the top of the stack.

```c
struct stack {
    int *contents;  //dynamic array of ints
    int top;       //stores the top of the stack
};
```

Begin by implementing the following stack operations:

```c
void init(struct stack *);  //initialize stack members
void push(struct stack *, int);  //grow contents to store int
int pop(struct stack *);   //shrink contents and return top int
```

When implementing these stack operations, don’t forget about the top member!

C Hints:
- Include <stdio.h> for printf/scanf
- Include <stdlib.h> for malloc/free and NULL
- Use the -> to access members from a struct pointer
- Use Monday's notes on the calendar page

You must compile with gcc!!!

```bash
gcc stack.c -o stack
```

Convince yourself and the TAs that the init(), push(), and pop() stack implementations are working!!!

(3 pts) A few more functions

```c
int peek(struct stack);  //Returns the top element in the stack
void destroy(struct stack *);  //destroy all elements in stack
bool empty(stack);  //return true if the stack is empty
int size(stack);    //return the number of elements in stack
```

Remember, you and your partner will not receive lab credit if you do not get checked off before leaving each lab. Once you have a zero on a lab, then it cannot be changed because we have no way of knowing if you were there or not!!!