Lab 7 (30 Points)

Objectives:

- Learn how to create and use a linked list ADT

Description:

1. Node * **createLinkedList**(int numOfNodes)
   - Create a linked list using **createNode**(key) where 1 ≤ **key** ≤ 10, using rand().
   - Insert the newly created Node at the head of the linked list.
   - Return: the head of the linked list.
2. Node * **createNode**(int num)
   - This function will create a Node struct using malloc() and initialize the Node **key** with **num** variable and Node **next** with NULL.
   - Return: the newly created Node.
3. Node * **findNode**(int searchKey, Node * head)
   - Input: an integer search key, and first node of the linked list.
   - Search the Node that contains the **searchKey**.
   - Return: Node pointer to the first found key node or **NULL** if not found.
4. void **displayList**(Node *head)
   - Input: first node of the linked list.
   - Display the keys in the linked list.
5. void **freeList**(Node *head)
   - Input: first node of the linked list.
   - Free all the Nodes in the linked list.
6. **Main** function
   - Create a linked list with 15 Nodes using **createLinkedList**().
   - Display your linked list with the **displayList**().
   - Print out the result as shown in the **Example** below and use what return back from **findNode**().
   - Free up the linked list using **freeList**().

```
typedef struct nodeL {
        int key;
        struct nodeL * next;
} Node;
```

Every user-defined function must have a comment describing:

- What function does;
- What parameter values are;
- What value it returns.

Example from the terminal window:

$ ./a.out

My linked list's keys: **2 3 4 5 10 7 8 9 4 3 1 2 3 4 6**

Enter an integer key between 1 to 10: **4**

The key **4** is found, and the next node of the key **4** Node is located at **0x7ffeefbff4d8**.

$ ./a.out

My linked list's keys: **2 3 4 5 6 7 8 9 3 9 4 5 6 7 2**

Enter an integer key between 1 to 10: **10**

The key **10** is not found!

Grading Criteria:

- Main program: 4 points
- createLinkedList function: 10 points
- createNode function: 4 points
- findNode function: 4 points
- displayList function: 4 points
- freeList function: 4 points

Note:

- If your code does not compile with **–Wall** and **–Werror**, you will receive a **zero** for this assignment.
- You need to finish at least **three** peer reviews within three days of this lab. Otherwise, you will get a 20% penalty.
- You will lose points if you don't have enough comments.