CSCD 240

A reference-based singly linked list is a basic data structure in computer science. Proficiency with linked lists is part of all CS tracks and represents a substantial portion of the Operating Systems class.

1. Create list.h and list.c that contains:
    - A structure called Node containing:
        - int data
        - struct Node * next

    - A structure called LinkedList containing:
        - Node * head;
        - int size

    - The following functions:
        - **LinkedList * linkedList()**
        - **void clear(LinkedList * myList)**
        - **int size(LinkedList * myList)**
        - **void addFirst(LinkedList * myList, int d)**
        - **void addLast(LinkedList * myList, int d)**
        - **int addDataByIndex(LinkedList * myList, int d, int index)**
            - Returns 1 on success
            - Returns 0 on failure (index out of bounds)
        - **void sort(LinkedList * myList)**
        - **Node* removeByIndex(LinkedList * theList, int index)**
            - Returns the Node removed
            - Returns NULL if nothing is removed
        - **int removeByValue(LinkedList * theList, int d)**
            - Returns 1 on success
            - Returns 0 on failure (item not found)
        - **printList(LinkedList * myList)**

2. I have provided an unchangeable main named cscd240Lab12.c. This is a simple C file meant to test the above functions.
3. You will need to figure out how to sort a linked list.

TO TURN IN

A zip file containing:

- All C/H files necessary to compile and run your program
- Include a Makefile with a target of lab12
- A valgrind run named lab12val.txt

.

Name your zip file your lastname first letter of your first name lab12.zip (Example: steinerslab12.zip)