CSCD 240
HW 6

PROGRAM SPECIFICATIONS

Create a singly linked list without a dummy head node that will represent an integer of any size (1 or more digits). Given cscd240_s13_hw6Tester.c develop the implementation of the "Huge Int" linked list. Your "Huge Int" should be able to handle both positive and negative numbers.

The node will contain the following, I have provided the .h file that contains the node. You may not change the node declaration. NOTE: You will not have global variables for this homework, and you will not have a size variable.

Node

Node contains the following:

- void * data;
- struct node * next;

cscd240_s13_hw6Tester.c

When the program starts up you will need to create 2 lists, one list for the first "Huge Int" and one list for the second "Huge Int". The user will enter a "Huge Int" that will be assigned to each number appropriately. You are guaranteed that user will only enter integers and possibly a negative sign as the first value.

I have provided a basic **readNumber** function. This function does not handle negative values. You can change readNumber, delete it, whatever, I don't care. (You **CANNOT** change cscd240_s13_hw6Tester.c) I was just trying to show you:

1) How to create the node
2) How to pass it to the addLast function.

Here is the function call from main (that is a lower case L not a one to start the variable name)

```
lOne = readNumber();
printNumber("One: ", lOne);

lTwo = readNumber();
printNumber("Two: ", lTwo);
```

printNumber simply prints the list name and then calls **printList**

Menu has the following choices:

1) Print the current “Huge Int” in each of the list
2) Allow the user to enter a new “Huge Int” for the first list
3) Allow the user to enter a new “Huge Int” for the second list
4) Add “Huge Int” one to “Huge Int” two using a plus method. The executeChoice method takes “Huge Int” two, “Huge Int” one, and then the function pointer to the the plus function.

   executeChoice(lTwo, lOne, &plus);

5) Subtract “Huge Int” two from “Huge Int one using a minus method. The executeChoice method takes “Huge Int” one, “Huge Int” two, and then the function pointer to the the minus function.

   executeChoice(lTwo, lOne, &minus);

   The executeChoice function will be the following:

```
Node * res = funcPtr(lTwo, lOne);
printNumber("The result is: ", res);
clearList(res);
```

   Notice the plus or minus function will return a Node *.

LIMITED SAMPLE RUN

Please enter a number 123456

One: 123456

Please enter a number 9876543210

Two: 9876543210

Select from the following:

1) Print Number 1 & Number 2

2) Change Number 1

3) Change Number 2

4) Add Number 2 to Number 1

5) Subtract Number 2 from Number 1

6) Quit

Choice --> 1

One: 123456

Two: 9876543210

Select from the following:
1) Print Number 1 & Number 2
2) Change Number 1
3) Change Number 2
4) Add Number 2 to Number 1
5) Subtract Number 2 from Number 1
6) Quit
Choice --> 4

9876543210 + 123456
The result is: 9876666666

Select from the following:
1) Print Number 1 & Number 2
2) Change Number 1
3) Change Number 2
4) Add Number 2 to Number 1
5) Subtract Number 2 from Number 1
6) Quit
Choice --> 5

123456 - 9876543210
The result is: -9876419754

Select from the following:
1) Print Number 1 & Number 2
2) Change Number 1
3) Change Number 2
4) Add Number 2 to Number 1
5) Subtract Number 2 from Number 1
6) Quit
Choice --> 6

IMPORTANT NOTES

- No global variables allowed
- No size variable allowed – If you want to know the size of the "Huge Int" then count the digits
- **I have the right to announce changes and updates to this homework in class. It is your responsibility to keep up to date on those changes.**
- If your code doesn't compile there will be no resubmissions (We will compile your code)
- If you submit the wrong hw (5 instead of 6) there will be no resubmissions

TO TURN IN

A **zip** file that contains:

- cscd240_s13_hw6Tester.c
- hw6.c
- hw6.h
- linkedlist.c
- linkedlist.h
- makefile – target of hw6
- output run named cscd240_s13_hw6out.txt
- valgrind run named cscd240_s13_hw6valgrind.txt
- Other C/H files you deem necessary

You should know the zip naming scheme by now. Last name first letter of first name hw6.zip (Example: steinershw6.zip)

Get started ASAP!