COP 4530
Project 1
Spring 2024

Instructions

For this programming project, you will implement a d*oubly linked list* to insert the integer data in increasing order and convert a list to a string.

You are not allowed to use the C++ Standard Library Lists or other sources to implement the linked list

Code testing- Use the intDList_driver.cpp and match the example interactions.

Submission

Submit a .zip file containing all your files with the format "`your login id_your group member login id_ assignment 1.zip`".
Your submission should contain the following files-

1.`IntDlist.cpp` (A template has been provided. You should only modify this file)
2.`IntDList.hpp` (Only modify if you add additional functions in intDList.cpp)

Abstract Class and Files

`void insertInOrder (int);(required function)`

Add nodes in an increasing order. So if we are adding nodes in the following manner-

```
myList.insertInOrder(9);
myList.insertInOrder(8);
myList.insertInOrder(0);
myList.insertInOrder(3);
```

And print the link list, it should return “`0389`”

`void` addToTail(int);(helper function)

Add a node with the input value as the tail(last) node

int deleteFromHead();(required function)

delete the head and return its value

int deleteFromTail();(required function)

Delete the tail and return its value

void deleteNode(int);(required function)

Delete node which contains the input integer value

string addToString() const;(required function)

This method returns the string of the ordered integers. Use `sstream` and `iomanip` (with argument to `setw` as 0). See `example.cpp` for details

Example interaction

Below are some examples of how your code should run(as shown in the driver file)

```
myList.insertInOrder(1);
myList.insertInOrder(2);
myList.insertInOrder(9);
myList.insertInOrder(4);
myList.insertInOrder(6);

myList.insertInOrder(3);

cout<< myList.addToString()<<endl;//should print "123469"


myList.deleteFromHead();

cout<< myList.addToString()<<endl;//should print "23469"


myList.deleteNode(4);

cout<< myList.addToString()<<endl;//should print "2369"


myList.deleteFromTail();

cout<< myList.addToString()<<endl;//should print "236"
```

Grading-

- Correct implementation of the required functions- **(4*20)=80**
- Correct Destructor implementation- **10**
- Documentation - **10** (1. Include comments detailing the logic behind the function implementations
  2. Add your name and your other group member's name at the top of your file as a comment)

Assumptions-

1.A value to be deleted is always present in the linked list.