

# CS235102

## Data Structure

### Homework 2

2016/11/01 10:00 am

~

2016/11/15 10:00 am  
(Hard Deadline)

# Target

- The target of this homework is to implement a linked list of integers
  - Each integer would be a node
  - Integers range from 0 to 99
  - Each node is unique, two duplicate integers won't exist at the same time
- E.g.
  - 25->3->42->77->10
  - 25->3->42->3->10 won't happened

# Target

- To implement the integer linked list, you are asked to implement 4 functions below
  - InsertBack (int data)
  - InsertAfter (int data1, int data2)
  - Delete (int data)
  - Reverse ()

# Target

- InsertBack (int data)

- Insert a data to the end of the linked list
- E.g. A->B → InsertBack (C) → A->B->C

- InsertAfter (int data1, int data2)

- Insert data2 after data1
- If data1 doesn't exist in the linked list, do nothing
- E.g. A->C → InsertAfter (A, B) → A->B->C
- E.g. A->C → InsertAfter (D, B) → A->C

# Target

- Delete (int data)

- Remove the data from the linked list
- If data doesn't exist in the linked list, do nothing
- E.g. A->B->C ➔ Delete (B) ➔ A->C
- E.g. A->B->C ➔ Delete (Y) ➔ A->B->C

- Reverse ()

- Reverse the linked list
- E.g. A->B->C ➔ Reverse () ➔ C->B->A

# File Structure

- **readonly**
  - **makefile**
  - **1.in** is a released test file
  - **main.cpp** contains a testing function
  - **class "Node"** represents the DS of a linked list node
  - **class "Chain"** represents a linked list
- **class "implementation"** contains your implementation

# Node

```
//=====THE-EDITS-OF-THIS-FILE-WILL-BE-DISCARDED=====
#ifndef Node_h
#define Node_h
#include <string>

//The data structure that present a node.
class Node
{
    friend class Chain;
public:
    //the pointer to next node in the chain
    Node *next;

    //stores the data
    int data;

    //constructor and destructor
    Node();
    Node(const int e, Node* next);
    ~Node();
};

#endif
//=====THE-EDITS-OF-THIS-FILE-WILL-BE-DISCARDED=====
```

# Chain

```
#include "Node.h"
using namespace std;

//It contains the fucntions that you have to override in implement.h/.cpp.
class Chain
{
public:
    virtual void InsertBack(int data);
    virtual void InsertAfter(int data1, int data2);
    virtual void Delete(int data);
    virtual void Reverse();

    string toString();
    Node* head = NULL;
};
#endif

//=====DO-NOT-MODIFY-THE-FILE=====
```



# Implement

- implement.h

```
class Implement : public Chain
{
public:
    // add your code here
    //-----
    //...
    //-----
};
```

- implement.cpp

```
#include "Implement.h"

// add your code here
//-----

//...

//-----
```

# 1.in

```
InsertBack 2
InsertBack 18
InsertBack 35
InsertBack 6
InsertBack 49
InsertAfter 6 7
InsertAfter 49 10
InsertAfter 1 20
Delete 35
Reverse
InsertBack 79
InsertBack 13
Delete 49
InsertAfter 10 11
Delete 7
Delete 6
Delete 88
Reverse
InsertBack 7
InsertBack 6
InsertBack 88
End 13->79->2->18->11->10->7->6->88
```

# Messages

- **[Undefined Chain::\*\*\*\*\*]**
  - The function is not implemented
- **[Wrong Answer]**
  - Wrong answer
- **[Accepted]**

# STL is not allowed

- `<list>` `<vector>` `<forward_list>` ... are not allowed
- If you try to include the above headers, your source files **WILL NOT** be compiled properly during TA's evaluation

# Submission

- make clean
  - Remove object files (\*.o) and the executable
- Archive your source codes (whole hw2 folder) into a zip file named [studentID]\_hw2.zip
  - E.g. 102062999\_hw2.zip
- Submit the zip file to ilms system **BEFORE** the deadline