

HW6 실습: 연결 리스트

(과제 아닙니다, 제출하지 않습니다.)

질문

pemds81718@gmail.com

다음을 작성하라.

(1) 실습 시작 전 만들어야 하는 파일들:

input.in

1 5 4 4 3 2

-1

1 3 2 2 5

-1

1 2

3 4

list.h

```
using namespace std;
struct Node
{
    Node(int d = 0, Node* l = NULL) : data(d), link(l) { }
    int data;
    Node * link;
};

struct IntList
{
    IntList() { last = first = NULL; }

    void Push_Back(int); // 리스트 뒤에 삽입 중복 허용
    void Push_Front(int); // 리스트 앞에 삽입 중복 허용
    void Insert(int); // 정렬되어 있다는 가정 하에 제 위치에 삽입
    void Delete(int); // 리스트의 원소 삭제
    Node *first; // 첫 노드를 가리킴
    Node *last; // 마지막 노드를 가리킴
};

ostream& operator<< (ostream&, IntList&);
```

hw6.cpp

```
#include<iostream>
#include"list.h"
using namespace std;

int main() {
    IntList il;
    int input;
    cout<<"====Input====" << endl;
    while(1) { // -1 을 받을 때 까지 반복
        cin>>input;
        if(input===-1) break;
        il.Insert(input);
        cout<<il;
    }
    cout<<"====Delete====" << endl;
    input=0;
    while(1) { // -1 을 받을 때 까지 반복
        cin>>input;
        if(input===-1) break;
        il.Delete(input);
        cout << il;
    }
    //Push_Front 와 Push_Back 은 2 회씩.
    cout<<"====Push_Front====" << endl;
    cin>>input; il.Push_Front(input); cout<<il;
    cin>>input; il.Push_Front(input); cout<<il;

    cout<<"====Push_Back====" << endl;
    cin>>input; il.Push_Back(input); cout << il;
    cin>>input; il.Push_Back(input); cout << il;
    return 0;
}
```

(2) list.cpp 의 비어있는 부분을 작성하시오

list.cpp

```
#include <iostream>
#include "list.h"
ostream& operator<<(ostream& os, IntList& il) {
    Node *ptr = il.first;
    while (ptr != NULL) {
        os << ptr->data << " ";
        ptr = ptr->link;
    }
    os << endl;
    return os;
}
void IntList::Push_Back(int e) {
    if (!first)
        first = last = new Node(e);
    else {
        last->link = new Node(e);
        last = last->link;
    }
}
void IntList::Push_Front(int e) { /* 작성 */ }

void IntList::Insert(int e) {
    //자유롭게 작성
    //Node 포인터 하나만으로도 리스트 순회 가능.
    if(!first){ /* 작성 */}
    else if(first->data >= e){ /* 작성 */}
    else{/* 작성 */}
}

void IntList::Delete(int e) {
    //자유롭게 작성
```

```
//Node 포인터 하나만으로도 리스트 순회 가능.
if(first){
    if(first->data == e){ /* 작성 */ }
    else{ /* 작성 */ }
}
}
```

(3) 결과

./hw6 input.in

```
=====Input=====
1
1 5
1 4 5
1 4 4 5
1 3 4 4 5
1 2 3 4 4 5
=====Delete=====
2 3 4 4 5
2 4 4 5
4 4 5
4 4
4 4
=====Push_Front=====
1 4 4
2 1 4 4
=====Push_Back=====
2 1 4 4 3
2 1 4 4 3 4
```