

Lab#6 Singly Linked List 연습

```
class Node {  
    private:  
        int data;  
        Node* next;  
        Node(int value) { data = value; next = 0; }  
        friend class List;  
};
```

```
class List {  
    private:  
        Node* head;  
    public:  
        List() { head = 0; }  
        ~List();  
        void insert(int);  
        void deleteN(int);  
        ..... }  
}
```

Main Program:

1. 메뉴 (1. insert 2. delete 3. search 4. **displayList** 5. FindMax 6. Quit)
2. Quit 선택 할 때까지 한 개씩의 메뉴 테스트 계속할 것
3. 매번 함수 실행 후 출력할 것. (아래 displaylist 이용가능)

Singly Linked-List ADT

- 1) Insert: 새로운 숫자 입력시, 오름차순으로 list에 insert 할것
if (head == NULL) => head = temp // 새로운 노드 생성
else if (new data < head->data) => head 이동
else { => 전체 리스트 검색 후 적합한 장소에 삽입한다.
while (p!=0 && p->data < head->data)
적합한 장소로 이동 //(ex. p, q 사용)
if (p != 0) insert here.
else 마지막에 삽입
- 2) Delete: delete 할 노드의 number 값으로 해당 노드를 찾아서 삭제할것
if (head == number) => move head & delete old head
else { // 전체리스트 검색후 해당노드를 삭제한다.
while (p!=0 && p->data < head->data)
적합한 장소로 이동 //(ex. p, q 사용)
if (p != 0) link & delete
else **"not found"** 출력.
- 3) 4) Search: //(예: 10 is found 또는 10 is not in the list) else "List empty"
찾고싶은 노드의 숫자(number)를 입력 받는다.

```

    if (head !=0) {
        while (p != 0 && p->data != number)
            // p 이동
        if (p!=0)    “found”
        else    “not found”
    }
    else    “List is empty”

```

4) FindMax: // 리스트에서 가장 큰 숫자를 출력한다 (findMax number)

```

p = max = head
while (p->next != NULL) {    // 다음노드 = p->next
    if (max 와 p 의 데이터 비교) // 다음 노드가 더크면
        max = 다음노드
    p 이동
}
Print max-number;

```

5) Quit: 리스트의 모든 노드들을 free 시켜야 한다.

**** displayList :** head 가 NULL 이면, print **"List empty"**
 아니면, 리스트의 전체 노드들을 출력

```

void List::displayList() {
    Node* p;
    if (head == 0)        cout << "List is empty!\n";
    else {
        p = head;
        cout << "--- List ---\n";
        for (int i = 1; p != 0; i++) {
            cout << setw(5) << i << "    :    " << p->data << endl;
            p = p->next;
        }
    }
}

```

<<TEST 예:>>

```
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 1
Input a number => 20
--- List ---
1 : 20
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 1
Input a number => 10
--- List ---
1 : 10
2 : 20
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 1
Input a number => 30
--- List ---
1 : 10
2 : 20
3 : 30
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 1
Input a number => 25
--- List ---
1 : 10
2 : 20
3 : 25
4 : 30
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 2
Input a number => 25
--- List ---
1 : 10
2 : 20
3 : 30
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 3
Enter a number => 20
20 is found.
--- List ---
1 : 10
2 : 20
3 : 30
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 4
Max number is : 30
--- List ---
1 : 10
2 : 20
3 : 30
Command : 1)insert, 2)delete, 3)search 4) findMax 5) quit => 5
```

이름#이메일#아이디#비밀번호#바탕화면#자루#조#자루#조2021#lab#lab6