

2020 DS 실습 13

박성진 조교

start date = 12 월 4 일(금)

due date = 12 월 18 일(금) 23 시 59 분 59 초 (※ 연장불가함)

Submit 방법 : *submit pem_ta hw13 분반*

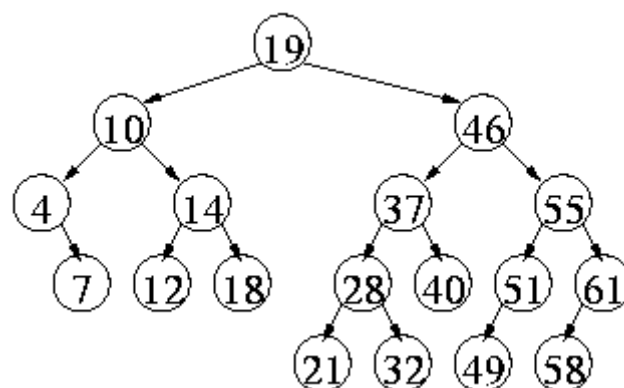
13번째 과제는 간단한 AVL Tree를 구현하는 과제입니다

1. 공지사항

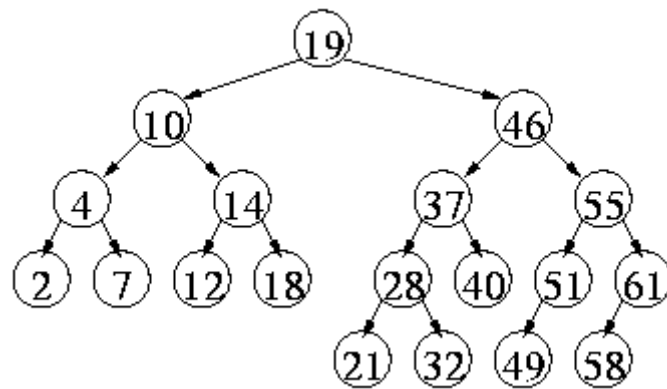
- 소스파일(헤더파일 포함), 결과보고서를 함께 리눅스에 submit(실행명은 hw13)
- 결과보고서는 Cheating 여부를 확인할 수 있는 수단으로서 pdf 로 최대한 자세히 작성
- C++로 작성(class)

2. AVL Tree

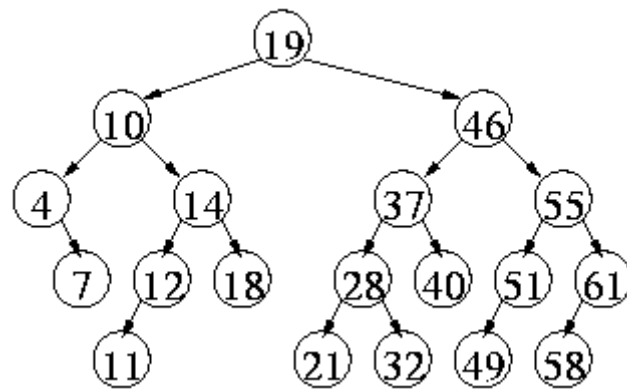
[ex1]



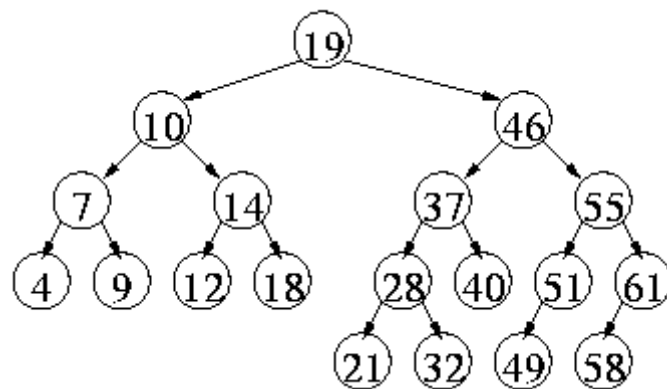
Insert 2



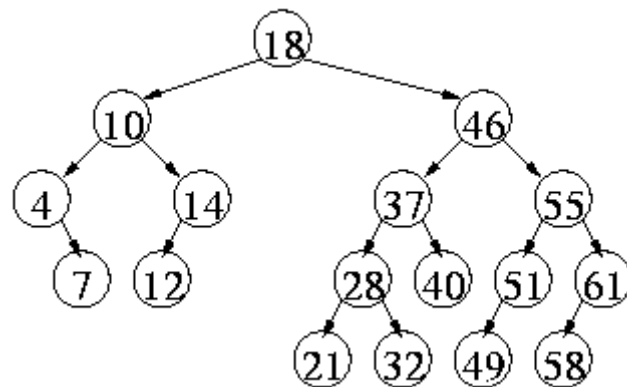
Insert 11



Insert 9



delete 19



3. 요구사항

1. 다음 조건(hw13.cpp)을 만족하는 AVL.h, AVL.cpp를 작성한다.

[hw13.cpp]

```
#include "AVL.h"
```

```
int main()
```

```
{
```

```
    nodeptr root;
```

```
    int a,choice,findele,delele;
```

```
    bstree bst;
```

```
    bool flag = false;
```

```
    root = NULL;
```

```
    bst.insert(19,root);
```

```
    bst.insert(10,root);
```

```
    bst.insert(46,root);
```

```

        bst.insert(4,root);          bst.insert(14,root);

        bst.insert(37,root);         bst.insert(55,root);         bst.insert(7,root);

        bst.insert(12,root);         bst.insert(18,root);

        bst.insert(28,root);         bst.insert(40,root);         bst.insert(51,root);

        bst.insert(61,root);         bst.insert(21,root);

        bst.insert(32,root);         bst.insert(49,root);         bst.insert(58,root);


while (flag==true)
{
    cout << "Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 :
exit) ";

    cin >> choice;

    switch (choice)
    {

        case 1:

            cout << "Enter node to search: ";

            cin >> findele;

            if (root != NULL)

                bst.Search(findele, root);

            break;


        case 2:

```

```

        cout << "Enter a new value: ";

        cin >> a;

        bst.insert(a, root);

        bst.Showresult(root);

        break;

    case 3:

        cout << "Enter node to delete: ";

        cin >> delele;

        bst.del(delele, root);

        bst.Showresult(root);

        break;

    case 4:

        if (root != NULL)

            bst.Showresult(root);

        break;

    case 0:

        cout << "\n\n\t\t\tThank your for using AVL tree program\n\n" <<
endl;

        flag = false;

        break;

    default:

```

```
        cout << "Sorry! wrong input\n" << endl;
        break;
    }
}

return 0;
}
```

2. 실행 예시

```
Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 : exit)
1
Enter node to search: 18
19 -> 10 -> 14 -> 18
Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 : exit)
2
Enter a new value: 2
19      left : 10      right : 46
10      left : 4       right : 14
46      left : 37      right : 55
4       left : 2       right : 7
14      left : 12      right : 18
37      left : 28      right : 40
55      left : 51      right : 61
2       left : empty   right : empty
7       left : empty   right : empty
12      left : empty   right : empty
18      left : empty   right : empty
28      left : 21      right : 32
40      left : empty   right : empty
51      left : 49      right : empty
61      left : 58      right : empty
21      left : empty   right : empty
32      left : empty   right : empty
49      left : empty   right : empty
58      left : empty   right : empty
Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 : exit)
3
Enter node to delete: 19
18      left : 10      right : 46
10      left : 4       right : 14
46      left : 37      right : 55
4       left : 2       right : 7
14      left : 12      right : empty
37      left : 28      right : 40
55      left : 51      right : 61
2       left : empty   right : empty
7       left : empty   right : empty
12      left : empty   right : empty
28      left : 21      right : 32
40      left : empty   right : empty
51      left : 49      right : empty
61      left : 58      right : empty
21      left : empty   right : empty
32      left : empty   right : empty
49      left : empty   right : empty
58      left : empty   right : empty
Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 : exit)
1
Enter node to search: 18
18
Enter the choice: (1 : search, 2 : add, 3 : delete, 4 : show, 0 : exit)
0

      Thank your for using AVL tree program

[pem_ta@localhost ~]$
```

3. 보고서 내용

1. class설계 내용 및 이유
2. 결과 값 및 결과 분석
3. 결과값 스크린 샷
4. AVL Tree 연산설명
 - 4종류의 회전 연산에 대한 설명 (그림 및 코드 주석 첨부)
5. 어려웠던 부분 등

4. 제출 파일

AVL.cpp, AVL.h, hw13.cpp

학번.pdf

5. 주의사항

- hw13.cpp 는 건드리지 말 것.
- 조교의 전달사항을 잘 따를 것. (미 출석하신 분은 따로 메일)
- 파일 이름은 정확하게 할 것.
- **Cheating 은 F.**
- 리눅스가 12 시간 이상 오류나지 않는다면 기간 연장은 없다.
- 기본적으로 메일로 제출된 과제는 읽지 않는다.
- 제출 마감시간 기준으로 4 시간 전부터 마감까지 제출이 불가능하다면, 이메일로 제출도 받는다.
- 하지만, 제출 마감시간 15 분 전부터 마감시간까지 제출이 가능했다면, 이메일로 제출된 것은 읽지 않는다.

● 질문

- a. pemta806@gmail.com 으로 질문한다.
- b. 영상통화 질문을 하고 싶다면 미리 메일로 스케줄을 잡는 것을 추천한다.
- c. 질문은 환영입니다. 단, 질문 사항을 명확하게 부탁드립니다. (코드는 주시고 왜 안되는지 질문해주세요...)