

Nonlinear Data Structure

<http://smartlead.hallym.ac.kr>

Instructor: **Jin Kim**
 010-6267-8189(033-248-2318)

jinkim@hallym.ac.kr

Office Hours:



Lab3(배열로 표현한 이진트리)

<http://smartlead.hallym.ac.kr>

Instructor: **Jin Kim**
010-6267-8189(033-248-2318)

Office Hours: jinkim@hallym.ac.kr



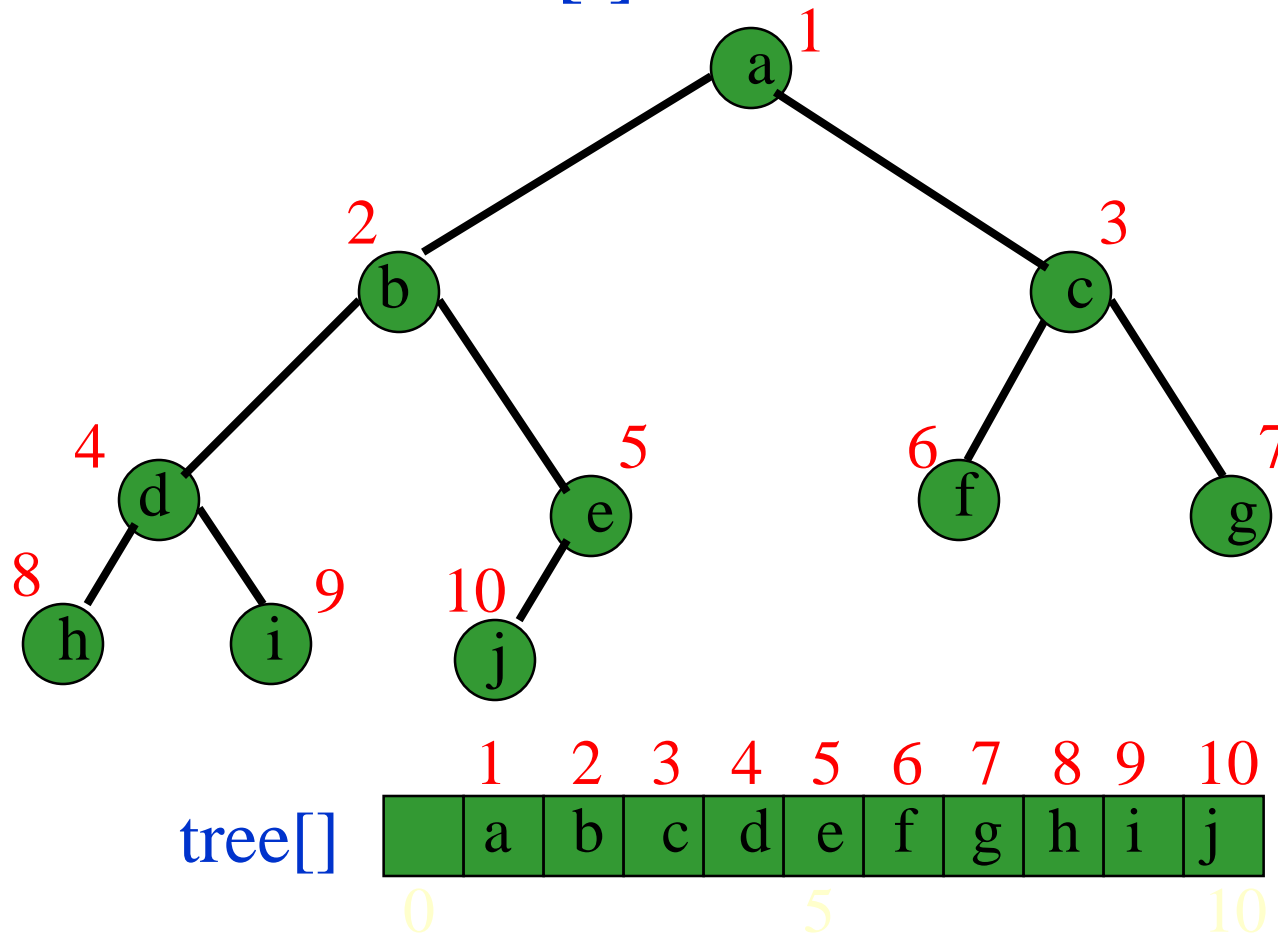
3주차 실습(배열로 이진트리구현)

- ◆ 다음 프로그램은 완전 이진 트리와 관련하여 배열로 구현한 것의 일부이다. 채워넣어라.



Array Representation(배열표현)

- ◆ Number the nodes using the numbering scheme for a full binary tree. The node that is numbered i is stored in `tree[i]`.



0. 주어진 프로그램에 `inorder()`, `postorder()`, `preorder()` method를 추가하라.
1. 주어진 프로그램에 특정원소(특정방번호입력받음)에 대한 조상(ancestors)들을 출력하는 부분 `pancestor()`를 추가하라
2. 주어진 프로그램에 특정원소(특정방번호입력받음)에 대한 후손(descendants)들을 출력하는 부분 `pdescendent()`를 추가하라.
3. 주어진 프로그램에 노드의 개수를 계산하는 `count method`를 추가하라.
4. 주어진 프로그램에 외부노드의 개수를 계산하는 `leafcount metho`를 추가하라.

프로그램을 학번+성명.zip으로 하여 업로드하라



참고 프로그램

```
class treearray {
    static int pancestor(char tree1[], int i) { // 조상들
        System.out.print(tree1[i]+" ");
        if(i==1){ return 0; }
        else { i=i/2;
            pancestor(tree1, i*2); // parent
        }
    }
    static int pdescendent(char tree[], int i) { // 후손들
        if(tree1[i]=="0"){ return 0; }
        else { System.out.print(tree1[i]+" ");
            pdescendent(tree1, i*2) // left child 왼쪽자식 한줄 채워라
            // right child 오른쪽 자식 한줄 채워라
        }
    }
}
```



```

static int inorder(char tree1[], int i) {
    if(tree1[i*2]!='\0'){ // 노드 i의 왼쪽 자식이 NULL 이 아니면 왼쪽
자식으로 이동한다.
        inorder(tree, i*2);
    }
    System.out.print(tree1[i]+" ");
    if(tree[i*2+1]!='\0') { // 노드 i의 오른쪽 자식이 NULL이 아니면 오른쪽
자식으로 이동한다.
        inorder(tree1, i*2+1);
    }
    return 0;
}

static int postorder(char tree[], int i) {
    // 채워라
}

static int preorder(char tree[], int i) {
    // 채워라
}

```



```

public static void main(String[] args) {
    char[] tree=new char[50];
    tree[1]='A';
    tree[2]='B';
    tree[3]='C';
    tree[4]='D';
    tree[5]='E';
    tree[6]='F';
    tree[7]='G';
    // 추가하라
    pancestor(tree, 5)// 5번방 원소의 선조를 찾아라
    pdescendent(tree, 3) //3번방 원소의 후손을 모두 찾아라
    inorder(tree,1);
    preorder(tree, 1);
    postorder(tree, 1);
} // end of main
} // end of class treearray

```



Upload your program(BT.java)(at
smartlead.hallym.ac.kr)

*.java 파일만을 업로드하라.
다른 파일들은 필요없다.

