

# 자료구조 실습 보고서

숙제명: [제11주]노드형태를이용한이진트리

제출일: 2018.05.15

학번/이름: 201404377 / 진승언

## <프로그램 설명서>

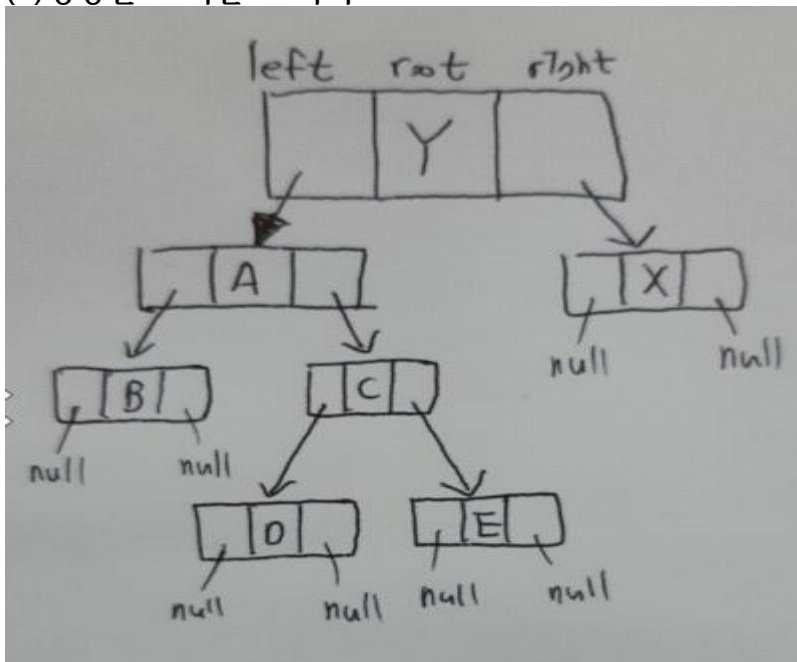
이진트리란 한 노드가 최대 두 개의 자식노드를 가지는 트리를 뜻한다. 보통 첫 번째 노드를 부모노드라고하며 자식노드는 왼쪽(left)과 오른쪽(right)으로 불린다. 트리구조를 가진 이진트리는 많은 순회 방법을 가지고 있다. 이진트리의 루트 노드에서 시작해서, 세 가지 주요 단계를 거치며 순회를 진행하는데, 이러한 과정은 재귀로 쉽게 설명 할 수 있다. 이는 밑에 그림과 같다.

전위 순회(preorder)	중위 순회(Inorder)	후회 순회(postorder)
<ol style="list-style-type: none"> <li>1. 노드 방문</li> <li>2. 왼쪽 서브트리 전위 순회</li> <li>3. 오른쪽 서브트리 전위 순회</li> </ol>	<ol style="list-style-type: none"> <li>1. 왼쪽 서브트리 중위 순회</li> <li>2. 노드 방문</li> <li>3. 오른쪽 서브트리 중위 순회</li> </ol>	<ol style="list-style-type: none"> <li>1. 왼쪽 서브 트리를 후위 순회</li> <li>2. 오른쪽 서브 트리 후위 순회</li> <li>3. 노드 방문</li> </ol>
(root, left, right) 루트 노드 우선	(left, root, right)	(left, right, root)

그림을 보면 알겠지만 2번의 순회와 한 번의 노드 방문이 있다. 여기서 노드 방문을 한 경우 그 노드를 출력해준다. 그리고 순회를 해야 하는 경우는 재귀로 왼쪽서브트리 혹은 오른쪽 서브트리로 재귀를 시켜준다. 위 그림을 참고하여 순회에 따라 코딩을 해주면 된다.

먼저 TestBinartTree 클래스에서 BinaryTree클래스를 사용해서 이진트리구조 데이터를 만들어주었다. 그 트리구조는 다음과 같다

### (1)생성된 트리를 그리기



```

public void preorderPrint() {
    if (root != null) {
        System.out.print(root);
    }
    if (left != null) {
        left.preorderPrint();
    }
    if (right != null) {
        right.preorderPrint();
    }
}

```

=> 전위순회이다. 전위순회는 VLR순회방법이다. 그러므로 처음에 노드를 방문해서 출력해주고 왼쪽노드를 순회하고 그다음 오른쪽 노드를 순회하면 된다. 순회는 재귀형식으로 해주면 된다. 그 코드가 위와 같다. 그럼 root노드 왼쪽 서브트리 오른쪽 서브 트리 순으로 순회하게 된다. 그리고 노드가 null이면 순회를 멈추게 해준다.

```

public void inorderPrint() {
    if (left != null) {
        left.inorderPrint();
    }
    if (root != null) {
        System.out.print(root);
    }
    if (right != null) {
        right.inorderPrint();
    }
}

public void postorderPrint() {
    if (left != null) {
        left.postorderPrint();
    }
    if (right != null) {
        right.postorderPrint();
    }
    if (root != null) {
        System.out.print(root);
    }
}

```

중위순회와 후위순회도 각각의 순회 순서에 맞게 위와 같이 코딩을 해주면 된다.

(2) 현재의 트리를 실습과 같이 배열로 구현할때 몇 번 배열에 넣어야 하는가

=>배열 인덱스를 번호로 편의상 번호로 부르면,

1번=Y / 2번=A / 3번=X / 4번=B / 5번=C / 10번=D / 11번=E

## <실행 결과 분석>

```

<terminated> TestBinaryTree [Java Application] C:\#f
treeY: [[ [B],A, [ [D],C, [E] ] ],Y, [X]]
YABCDEX
BADCEYX
BDECAXY

```

순회가 잘 되었음을 볼 수 있다.