



알고리즘 2주차

이진트리

MMC 연구실

박사 과정 문희찬

조교 소개

- 문희찬
- 컴퓨터공학과 대학원 석사과정
- MMC연구실 (A1406)
- HCMoon@hallym.ac.kr



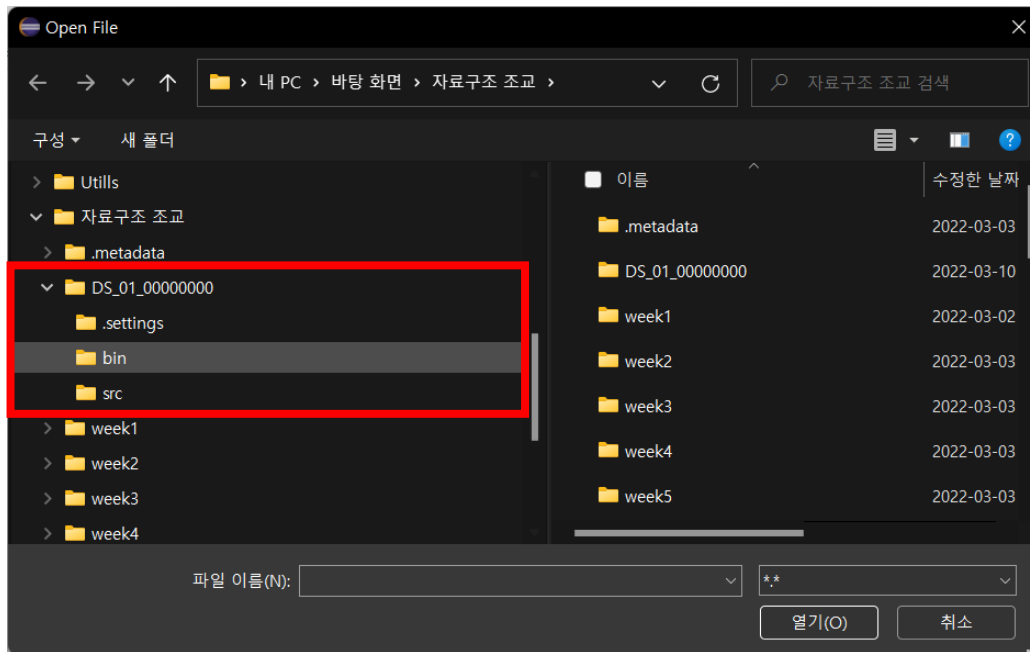
실습 수업 진행 방식

- 쉬는 시간 없이 1시간 30분 수업 (화장실 자유롭게 다녀오세요)
- 출석체크 : 수업 시작, 수업 끝날 때 체크
- 수업 시작 30분 뒤부터, 확인 문제를 해결한 학생은 검사 받고 퇴실
- 과제 진행 중 모르는 부분은 메일로 질문

과제 설명

- 알고리즘 수업은 Eclipse를 사용하여 코드를 작성합니다.
- 확인 문제 및 과제를 전부 해결하여 제출해주세요.
- 과제 제출 시 **프로젝트 폴더를 압축**해서 제출합니다.
- 과제의 채점은 프로젝트의 실행 결과를 기준으로 점수를 매깁니다.
- 컨닝 금지, 모르는 것이 있으면 저에게 질문해주세요.
(메일 주소 확인)

과제 제출 방법



- 프로젝트 폴더를 압축하여 제출

- 프로젝트이름 : AL_(주차)_(학번)

예) AL_02_00000000

- *.java파일만 제출하면 안됩니다.

- 제출양식을 반드시 지켜주세요!

확인문제 (binarytree)

Package Name : binarytree

Class Name : BTNode

```
public class BTNode {  
    String data;  
    BTNode Lchild;  
    BTNode Rchild;  
  
    public BTNode(String dt) {  
        data = dt;  
        Lchild = null;  
        Rchild = null;  
    }  
  
    public BTNode(BTNode lc, String dt, BTNode rc) {  
        data = dt;  
        Lchild = lc;  
        Rchild = rc;  
    }  
}
```

확인문제 (binarytree)

Package Name : binarytree

Class Name : BinaryTree

```
public class BinaryTree {
    BTNode root;

    public BinaryTree() {
        root = null;
    }
    public BinaryTree(String data) {
        root = new BTNode(null, data, null);
    }
    public BinaryTree(BinaryTree lTree, String data, BinaryTree rTree) {
        root = new BTNode(lTree.root, data, rTree.root);
    }

    public boolean isEmpty() {
        // 현재 Tree에 노드가 없다면 True 반환
    }

    public BinaryTree leftSubTree() {
        // 현재 Tree의 왼쪽 서브트리를 반환
    }

    public BinaryTree rightSubTree() {
        // 현재 Tree의 오른쪽 서브트리를 반환
    }

    public String rootData() {
        // root Node의 데이터를 반환
    }
}
```

확인문제 (binarytree)

Package Name : binarytree

Class Name : BinaryTree

```
public void printTree() {  
    printTree(root);  
}  
  
private void printTree(BTNode t) {  
    // 중위 순회를 이용하여 이진트리에 입력된 수식을 출력  
}
```


확인문제 (binarytree)

Package Name : binarytree

Class Name : BinaryTreeTest

```
public class BinaryTreeTest {  
    public static void main(String[] args) {  
        BinaryTree btree;  
        BinaryTree ltree;  
        BinaryTree rtree;  
  
        rtree = new BinaryTree("1");  
        ltree = new BinaryTree("2");  
        btree = new BinaryTree(ltree, "+", rtree);  
        ltree = btree;  
        rtree = new BinaryTree("3");  
        btree = new BinaryTree(ltree, "-", rtree);  
  
        btree.printTree();  
    }  
}
```

<terminated> BinaryTreeTest [Java Application]

2 + 1 - 3

실습 과제

1. 이진트리 수식 계산



과제 1

Package Name : binarytree

Class Name : BinaryTree

```
public int calculate() {  
    return theCalculate(root);  
}  
  
private int theCalculate(BTNode t) {  
    // 재귀 알고리즘을 이용하여 이진 트리에 입력된 수식을 계산  
}
```



과제 1

Package Name : binarytree
Class Name : BinaryTreeTest

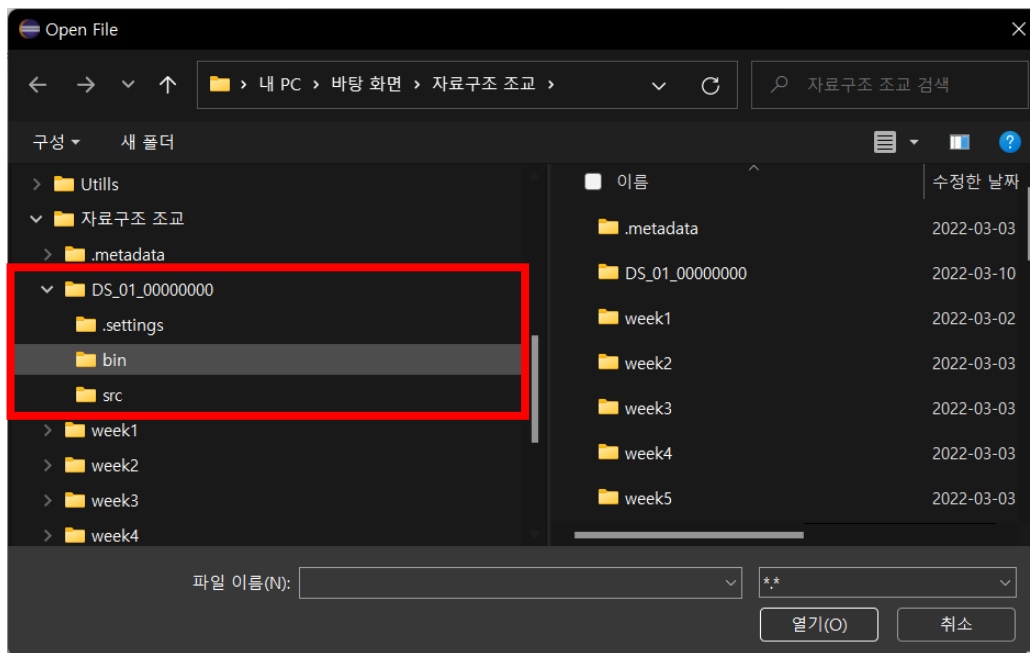
```
public class BinaryTreeTest {  
    public static void main(String[] args) {  
        BinaryTree btree;  
        BinaryTree ltree;  
        BinaryTree rtree;  
  
        rtree = new BinaryTree("1");  
        ltree = new BinaryTree("2");  
        btree = new BinaryTree(ltree, "+", rtree);  
        ltree = btree;  
        rtree = new BinaryTree("3");  
        btree = new BinaryTree(ltree, "-", rtree);  
  
        btree.printTree();  
        System.out.println(" = " + btree.calculate());  
    }  
}
```



<terminated> BinaryTreeTest [Java Application]

2 + 1 - 3 = 0

과제 제출 방법



- 프로젝트 폴더를 압축하여 제출

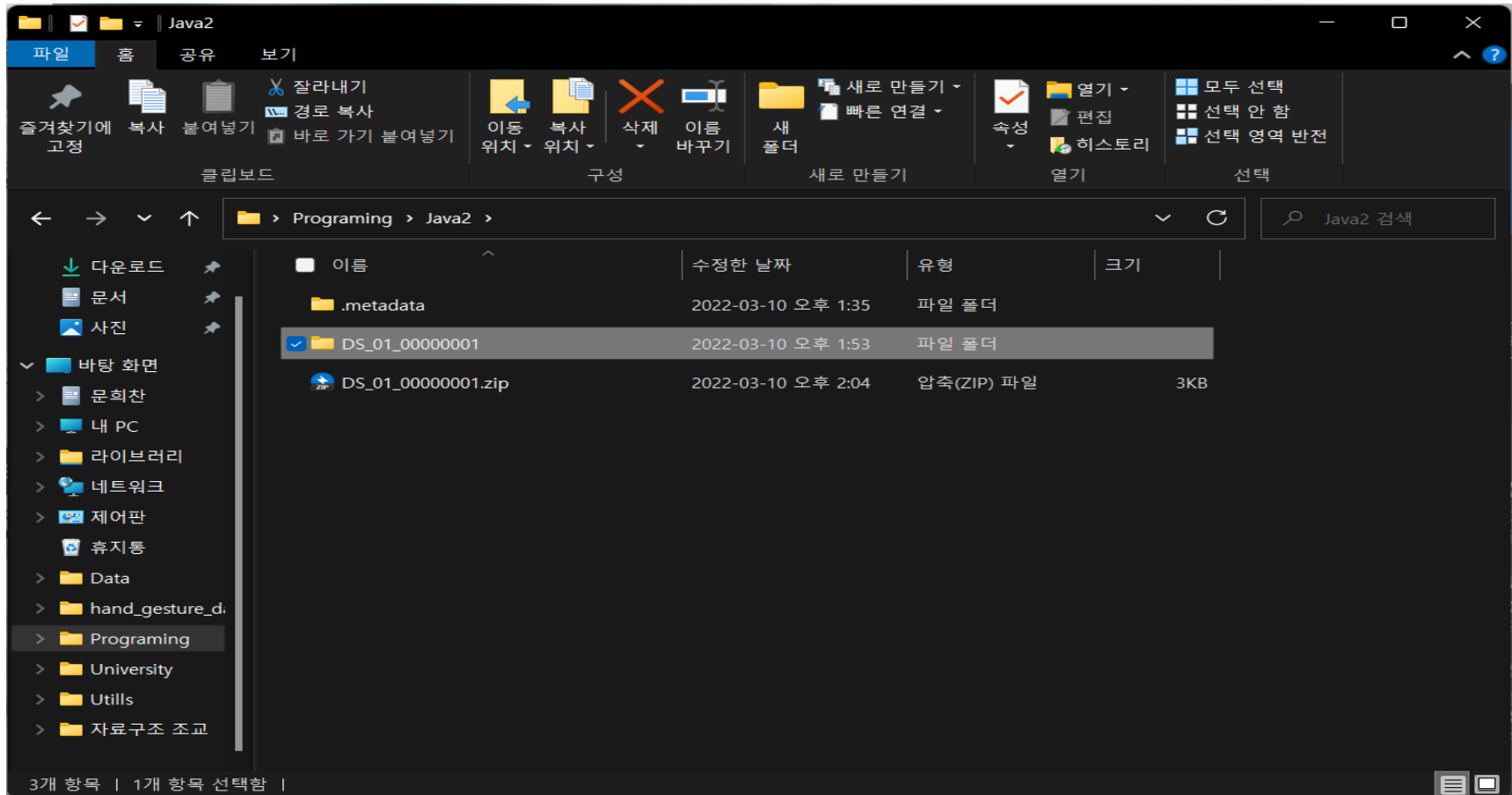
- 프로젝트이름 : AL_(주차)_(학번)

예) AL_01_00000000

- *.java파일만 제출하면 안됩니다.

제출양식을 반드시 지켜주세요

과제 제출 방법



- 반드시 **프로젝트 폴더를 압축**하여 제출