



# 자료구조 9주차

다항식

MMC 연구실

석사 과정 강민제

## 조교 소개

- 강민제
- 컴퓨터공학과 대학원 석사과정
- MMC연구실 (A1406)
- rkdalswp29@gmail.com



## 실습 수업 진행 방식

- 확인 문제 풀이
- 확인 문제를 해결한 학생은 검사 받고 퇴실



## 과제 설명

- 자료구조 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.

# 확인문제 1 (계수 다항식)

Package Name : poly

Class Name : Poly

```
public class Poly {
    private static final int MAX = 30;
    private int[] coef;
    private int degree;

    public Poly() {
        coef = new int[MAX + 1];
        for(int i = 0; i <= MAX; i++)
            coef[i] = 0;
        degree = -1;
    }
    public boolean isPZero() {
        return (degree == -1);
    }

    public int coef(int e) {
        return coef[e];
    }

    public int maxExp() {
        return degree;
    }

    public void printResult() {
        for(int i = degree; i >= 0; i--) {
            if(i == 0) {
                System.out.println(coef[i]);
                break;
            }
            if(coef[i] != 0)
                System.out.print(coef[i] + "x^" + i + " + ");
        }
    }
}
```

```
public boolean addTerm(int c, int e) {
    // e의 차수에 계수 c를 삽입, degree 변경
    // 단, coef[e]에 이미 계수가 존재하는 경우에는 계수 c를 삽입하지않고 false 리턴
}

public boolean delTerm(int e) {
    // e의 차수를 갖는 계수를 다항식에서 삭제 (계수를 0으로 만들), degree 변경
}

public Poly sMult(int c, int e) {
    // 현재 다항식에  $cx^e$ 를 곱하여 새 다항식을 반환
}
public Poly polyAdd(Poly poly1) {
    // 현재 다항식과 다른 다항식 poly1을 더한 결과 다항식을 반환
}

public Poly polyMult(Poly poly1) {
    // 현재 다항식과 다른 다항식 poly1을 곱한 결과 다항식을 반환
}
```

poly/Poly.txt 파일 참고

polyMult -> sMult, polyAdd 이용

## 확인문제 1 (계수 다항식)

Package Name : poly

Class Name : Main

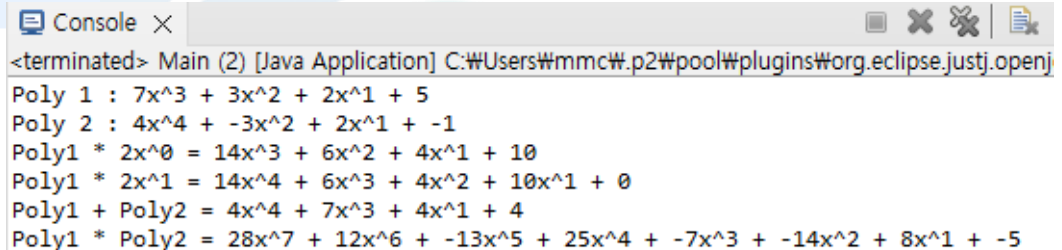
```
package poly;
public class Main {
    public static void main(String[] args) {
        Poly poly1 = new Poly();
        poly1.addTerm(5, 0);
        poly1.addTerm(2, 1);
        poly1.addTerm(3, 2);
        poly1.addTerm(7, 3);
        System.out.print("Poly 1 : ");
        poly1.printResult();

        Poly poly2 = new Poly();
        poly2.addTerm(-1, 0);
        poly2.addTerm(2, 1);
        poly2.addTerm(-3, 2);
        poly2.addTerm(4, 4);
        System.out.print("Poly 2 : ");
        poly2.printResult();

        System.out.print("Poly1 * 2x^0 = ");
        poly1.sMult(2, 0).printResult();
        System.out.print("Poly1 * 2x^1 = ");
        poly1.sMult(2, 1).printResult();

        System.out.print("Poly1 + Poly2 = ");
        poly1.polyAdd(poly2).printResult();

        System.out.print("Poly1 * Poly2 = ");
        poly1.polyMult(poly2).printResult();
    }
}
```



Console X

<terminated> Main (2) [Java Application] C:\Users\mmc\p2\pool\plugins\org.eclipse.justj.openj

Poly 1 :  $7x^3 + 3x^2 + 2x^1 + 5$   
Poly 2 :  $4x^4 + -3x^2 + 2x^1 + -1$   
Poly1 \*  $2x^0 = 14x^3 + 6x^2 + 4x^1 + 10$   
Poly1 \*  $2x^1 = 14x^4 + 6x^3 + 4x^2 + 10x^1 + 0$   
Poly1 + Poly2 =  $4x^4 + 7x^3 + 4x^1 + 4$   
Poly1 \* Poly2 =  $28x^7 + 12x^6 + -13x^5 + 25x^4 + -7x^3 + -14x^2 + 8x^1 + -5$

poly/Main.txt 파일 참고

# 과제 1 (계수-지수 다항식)

Package Name : poly2

Class Name : Poly\_Term

```
public class Poly_Term {
    private static final int MAX = 10;
    private Term[] terms;
    private int noOfTerms;

    public Poly_Term() {
        terms = new Term[MAX];
        noOfTerms = 0;
    }

    public boolean isPZero() {
        if (noOfTerms == 0)
            return true;
        else
            return false;
    }

    public int coef(int e) {
        for (int i = 0; i < noOfTerms; i++) {
            if (terms[i].getExp() == e)
                return terms[i].getCoef();
        }
        return 0;
    }

    public int maxExp() {
        if (noOfTerms == 0)
            return -1;
        else
            return terms[0].getExp();
    }
}
```

```
public void addTerm(int c, int e) {
    // 주어진 계수 c와 지수 e를 사용하여 새로운 항을 다항식에 추가하는 메소드
    // 이 메소드는 새로운 항을 알맞은 위치에 추가하고 noOfTerms를 업데이트 해야 함
    // tip) 새로운 항의 지수 e보다 작은 지수를 가진 항들을 한 칸씩 오른쪽으로 이동
}

private int findIndex(int e) {
    for (int i = 0; i < noOfTerms; i++)
        if (terms[i].getExp() == e)
            return i;
    return -1;
}

public void delTerm(int e) {
    int index = findIndex(e);
    // 주어진 지수 e를 갖는 항을 다항식에서 삭제하는 메소드
    // 해당 항을 찾아 배열에서 제거하고, 이후 항들을 앞으로 이동시켜 빈 공간을 없애야 함
    // 삭제 후 noOfTerms를 업데이트 해야 함
}

public void printResult() {
    for (int i = 0; i < noOfTerms; i++) {
        if (i == noOfTerms - 1) {
            System.out.println(terms[i].getCoef());
            break;
        }
        System.out.print(terms[i].getCoef() + "x^" + terms[i].getExp() + " + ");
    }
}
```

poly2/Poly\_Term.txt 파일 참고

## 과제 1 (계수-지수 다항식)

Package Name : poly2

Class Name : Term

```
package poly2;

public class Term {
    private int coef;
    private int exp;

    public Term(int coef, int exp) {
        this.coef = coef;
        this.exp = exp;
    }

    public int getCoef() {
        return coef;
    }

    public int getExp() {
        return exp;
    }
}
```

poly2/Term.txt 파일 참고

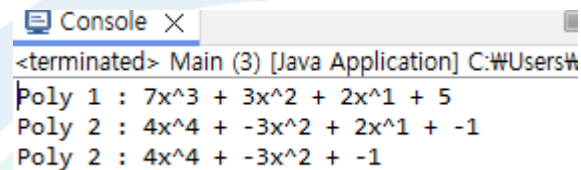


## 과제 1 (계수-지수 다항식)

Package Name : poly2

Class Name : Main

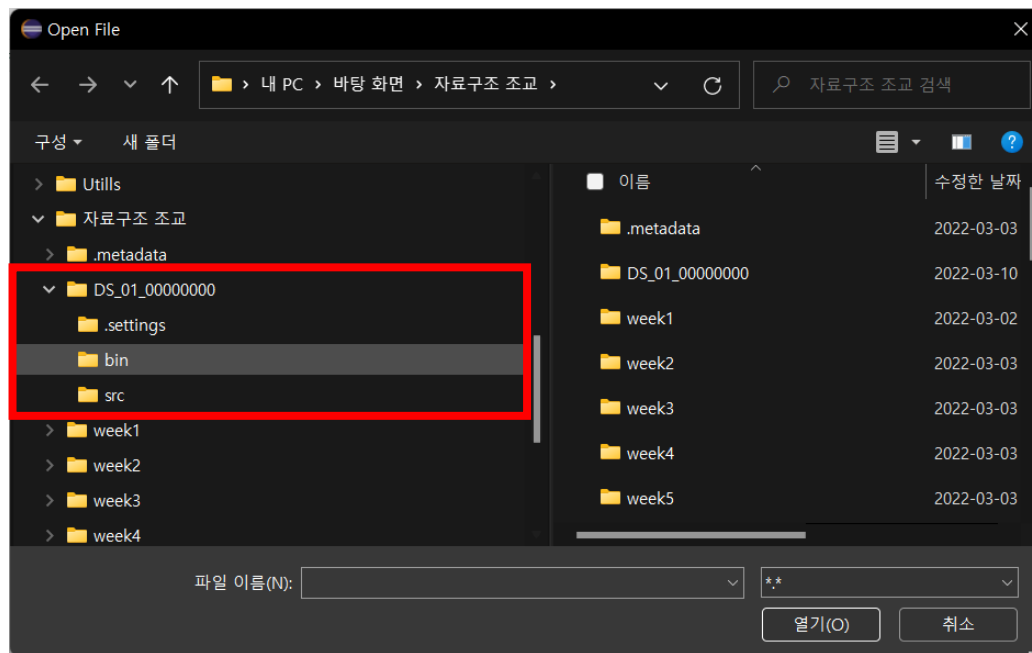
```
public class Main {  
  
    public static void main(String[] args) {  
        Poly_Term poly1 = new Poly_Term();  
        poly1.addTerm(5,0);  
        poly1.addTerm(3,2);  
        poly1.addTerm(2,1);  
        poly1.addTerm(7,3);  
        System.out.print("Poly 1 : ");  
        poly1.printResult();  
  
        Poly_Term poly2 = new Poly_Term();  
        poly2.addTerm(4, 4);  
        poly2.addTerm(2, 1);  
        poly2.addTerm(-3, 2);  
        poly2.addTerm(-1, 0);  
        System.out.print("Poly 2 : ");  
        poly2.printResult();  
        poly2.delTerm(1);  
        System.out.print("Poly 2 : ");  
        poly2.printResult();  
    }  
}
```



Console X  
<terminated> Main (3) [Java Application] C:\Users\#  
Poly 1 : 7x^3 + 3x^2 + 2x^1 + 5  
Poly 2 : 4x^4 + -3x^2 + 2x^1 + -1  
Poly 2 : 4x^4 + -3x^2 + -1

poly2/Main.txt 파일 참고

# 과제 제출 방법



- 프로젝트 폴더를 압축하여 제출

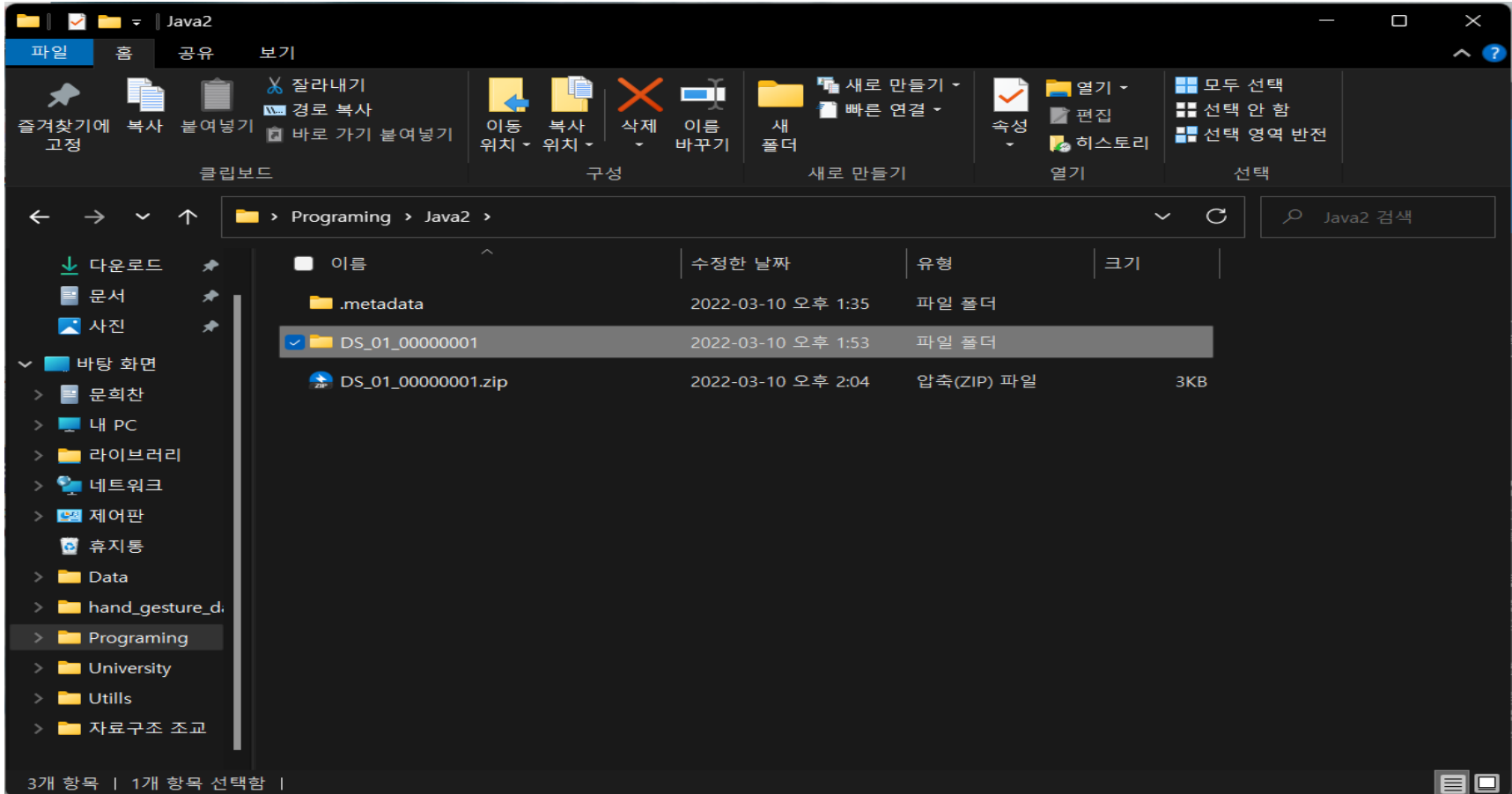
- 프로젝트이름 : DS\_(주차)\_(학번)

예) DS\_09\_00000000

- \*.java파일만 제출하면 안됩니다.

**제출양식을 반드시 지켜주세요**

# 과제 제출 방법



- 반드시 **프로젝트 폴더를 압축**하여 제출