



객체 지향 언어 및 실습



학번 : 2016110056

학과 : 불교학부

이름 : 박승원

날짜 : 2017년 4월 7일

문제 1 (12점, 부분점수 없음)

다음 아래의 질문에 답하라.

- (1) 생성자는 어떤 용도로 쓰이는가? 객체를 생성한다.
- (2) 중복 정의된 메소드들은 어떻게 구별되어서 호출되는가? 인자의 형을 검사한다.
- (3) 키워드 this 는 무엇을 가리키는가? 클래스가 자기 자신을 참조한다.
- (4) 정적 변수와 인스턴스 변수의 차이점은 무엇인가? 정적변수는 클래스의 초기화가 없이 모든 클래스가 공통으로 가지는 변수이고, 인스턴스 변수는 클래스의 각 인스턴스가 따로 가지는 변수이다.
- (5) 객체가 매개 변수로 전달될 때는 어떤 값이 전달되는가? 객체의 참조값이 전달되므로, 복사되어 전달되는 경우와 달리 객체의 내용을 변경할 수 있다.
- (6) 왜 정적 메소드는 인스턴스 변수를 참조할 수 없는가? 정적메소드는 인스턴스 생성없이 사용할 수 있는 메소드이므로, 아직 생성이 안된 인스턴스를 참조해서는 안된다.

문제 2 (8점, 부분점수 없음)

- 다음 코드에서 오류를 찾고 수정하라.

```
(1)
public class Point {
    private int x, y;
    public void Point(int x, int y) {
        this.x = x;
        this.y = y;
    }
}
```

메소드의 오버로딩은 인자의 형이 달라야 하므로

```
(2) 사용되지 않는 인자라도 하나 넣어주면 된다.
public class MyMath {
    public int getRandom() {
        return (int) Math.random();
    }
    public double getRandom() {
        return Math.random();
    }
}
```

문제 2 (Cont.)

- 다음 코드에서 오류를 찾고 수정하라.

```
(3)
public class MyClass {
    static private String getName() {
        return "MyClass";
    }
    public static String getClassName() {
        return getName();
    }
}
```

문제 3 (20점)

- 강아지를 나타내는 Dog 이라는 이름의 클래스를 설계한다. Dog 클래스는 다음과 같은 필드를 가져야 한다. 각 필드는 getter 와 setter가 있어야 한다.
- name: 강아지의 이름, 전용 멤버
- breed: 강아지의 종류, 예를 들면 “요크셔테리어”, 공용 멤버
- age: 강아지의 나이, 전용 멤버

Dog 클래스는 다음과 같은 생성자를 가져야 한다. 초기화되지 않은 필드를 null 이나 0 으로 초기화하라.

- Dog(String name, int age): 강아지의 이름과 나이를 초기화
- Dog(String name, String breed, int age): 강아지의 이름, 종류, 나이를 초기화

```
class Dog {  
    private String name;  
    public String breed;  
    private int age;  
  
    String getName() { return name; }  
    void setName(String name) { this.name = name; }  
    int getAge() { return age; }  
    void setAge(int age) { this.age = age; }  
  
    public Dog(String name, int age) {  
        this.name = name;  
        this.age = age;  
        this.breed = null;  
    }  
  
    public Dog(String name, String breed, int age) {  
        this.name = name;  
        this.age = age;  
        this.breed = breed;  
    }  
}  
  
public class problem3  
{  
    static public void main(String[] args) {  
        Dog d = new Dog(" 멍멍이 ", " 진돗개 ", 3);  
        d.setAge(5);  
    }  
}
```

문제 4 (35점)

- 다양한 행렬을 표현하는 클래스를 형성해보자.
- 행렬을 표현하는 클래스 Matrix의 필드 명세는 아래와 같다.

필드(Field)

행 : row(private), 열 : col(private)

배열 요소 : array(public)

배열의 총 개수 : array_number(static public)

// 2016110056 박승원

```
class Matrix {  
    private int row = 0;  
    private int col = 0;  
    public int[][] array = null;  
    static public int array_number=0;  
  
    void setRow(int row) { this.row = row; }  
    int getRow() { return row; }  
    void setCol(int col) { this.col = col; }  
    int getCol() { return col; }  
    static void getArrayNumber() {  
        System.out.println(" 총 행렬의 갯수는 " + array_number + " 입니다 .");  
    }  
  
    Matrix(int col, int row) {  
        this.col = col;  
        this.row = row;  
        array = new int[col][row];  
        array_number++;  
    }  
}
```

```

    }
    private int rowXcol(int [][] a, int [][] b, int r, int c, int n) {
        int sum = 0;
        for(int i=0; i<n; i++) sum += a[i][r] * b[c][i];
        return sum;
    }
    public void show() {
        for(int i=0; i<row; i++) {
            for(int j=0; j<col; j++) System.out.print(array[j][i] + " ");
            System.out.println();
        }
        getArrayNumber();
    }
}

```

```

Matrix multiply(Matrix r) {
    Matrix m = new Matrix(r.col, row);
    for(int i=0; i<r.col; i++) for(int j=0; j<row; j++)
        m.array[j][i] = rowXcol(array, r.array, i, j, col);
    return m;
}

```

```

Matrix multiply(int n) {
    Matrix m = new Matrix(col, row);
    for(int i=0; i<col; i++) for(int j=0; j<row; j++)
        m.array[i][j] = n * array[i][j];
    return m;
}

```

```

Matrix summation(Matrix r) {
    Matrix m = new Matrix(col, row);
    for(int i=0; i<col; i++) for(int j=0; j<row; j++)
        m.array[i][j] = array[i][j] + r.array[i][j];
    return m;
}

```

```

Matrix transpose() {
    Matrix m = new Matrix(row, col);
    for(int i=0; i<col; i++) for(int j=0; j<row; j++)
        m.array[j][i] = array[i][j];
    return m;
}

```

```

}

```

```

public class problem4 {
    public static void main(String[] args) {

```

```

Matrix m1 = new Matrix(3,2);
for(int i=0,k=0; i<3; i++) for(int j=0; j<2; j++) m1.array[i][j] = k++;
m1.show();
Matrix m2 = new Matrix(2,3);
for(int i=0,k=0; i<2; i++) for(int j=0; j<3; j++) m2.array[i][j] = k++;
m2.show();
Matrix m3 = m1.multiply(m2);
System.out.println ( " 곱 " );
m3.show();
Matrix m4 = m3.transpose();
System.out.println ( " 전치행렬 " );
m4.show();
Matrix m5 = m4.multiply(4);
System.out.println ( " 상수배 " );
m5.show();
Matrix m6 = m5.summation(m4);
System.out.println ( " 덧셈 " );
m6.show();
System.out.println ( " 행렬을      입력하세요  ." );
System.out.println ( " 가로의      크기를      입력하세요  ." );
int w = JavaUtil . getInt () ;
System.out.println ( " 세로의      크기를      입력하세요  ." );
int h = JavaUtil . getInt () ;
System.out.println ( " 행렬을      입력하세요  ." );
Matrix m7 = new Matrix(w, h);
for(int i=0; i<h; i++) for(int j=0; j<w; j++)
    m7.array[j][i] = JavaUtil . getInt () ;
m7.show();

```

```

    }

```

```

}

```

```
zezeon@ubuntuZ: ~/Programming/java
SwingExample.png problem1.png problem7.png
algo.class problem2.class problem8.class
65.43%
zezeon@ubuntuZ:~/Programming/java$ javac problem4.java
problem4.java:2: error: package util does not exist
import util.*;
^
1 error
zezeon@ubuntuZ:~/Programming/java$ javac problem4.java
zezeon@ubuntuZ:~/Programming/java$ java problem4
0 2 4
1 3 5
총 행렬의 갯수는 1입니다.
0 3
1 4
2 5
총 행렬의 갯수는 2입니다.
급
10 28
13 40
총 행렬의 갯수는 3입니다.
전치행렬
10 13
28 40
총 행렬의 갯수는 4입니다.
상수배
40 52
112 160
총 행렬의 갯수는 5입니다.
덧셈
50 65
140 200
총 행렬의 갯수는 6입니다.
행렬을 입력하세요.
가로의 크기를 입력하세요.
2
세로 크기를 입력하세요.
2
행렬을 입력하세요.
1
2
3
4
1 2
3 4
총 행렬의 갯수는 7입니다.
zezeon@ubuntuZ:~/Programming/java$
```

문제 5 (25점)

- 다음 문자열이 파일에 몇 개 있는지 살펴보자.

- 입력: 읽을 파일명, 찾을 문자열 배열
- 출력: 각각의 문자열이 몇 개 있는지 출력

- Ex) 입력] input txt : abcdefg aaA A
찾을 문자열 : a, A, cd
출력] a는 3개, A는 2개, cd는 1개

문제 5 (25점)

- 다음 문자열이 파일에 몇 개 있는지 살펴보자.

1. 입력 받은 문자열을 뒤집은 형태
ex) abc -> cba

- 추가로 특정 문자열이 나타나는 문장을 출력하라.

- Ex) 입력] txt : abcdefg aaA A
찾을 문자열 : a, A
출력] 3개의 a : abcdefg, aaA, aaA
2개의 A : aaA, A

문제 5 (25점)

- 파일에 특정 문자열이 몇 개 있는지 살펴보자.

- 입력 : 읽을 파일명, 찾을 문자열 배열
- 출력 : 각각의 문자열이 몇 개 있는지 출력

- Ex) 입력] input txt : abcdefg aaA A

찾을 문자열 : a, A, cd

출력] a는 3개, A는 2개, cd는 1개

// 2016110056 박승원

```
import java.io.*;
import java.util.*;
public class problem5 {
    static String trim(String s, int pos) { // return the part pos is located
        String r = "";
        for(int i=0; i<s.length(); i++) {
            if(s.charAt(i) == ' ') {
                if(i<pos) r = "";
                else return r;
            } else r += s.charAt(i);
        }
        return r;
    }

    public static void main(String[] ar) throws java.io.IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        System.out.print(" 찾을 파일명을 입력하세요 :");
        String file = br.readLine();
        String s, t="";
        try {
```

```

        BufferedReader in = new BufferedReader(new FileReader( file ));
        while((s = in.readLine()) != null) t += s;
        in.close();
    } catch(IOException e) {
        System.out.println(" 그런 파일은 없습니다 .");
        return;
    }
    System.out.print(" 찾을 문자열을 입력하세요 (comma로 구문을 띄우세요) :");
    String search = br.readLine();
    br.close();
    String[] strim = new String[10];
    String[] to_find = search.split(",");

    //for(int i=0; i<t.length(); i++) t.charAt(i);
    for(String s2 : to_find) {
        int ppos=-1, occur=0;
        while((ppos = t.indexOf(s2, ppos+1)) != -1) {
            strim[occur] = trim(t, ppos);
            occur++;
        }
        System.out.print(s2 + " 는 " + occur + " 개 : ");
        for(int i=0; i<occur; i++) System.out.print(strim[i] + " ");
        System.out.println();
    }
}
}

```

```
o.java report.tex problem5.java problem7.java problem6.java JavaUtil.java problem4.java problem2.java
(// \)
(// \)
public class problem4 {
    public static void main(String[] args) {
        Matrix m1 = new Matrix(3, 3);
        for(int i=0; i<3; i++) for(int j=0; j<3; j++) m1.array[i][j] = k++;
        m1.show();
        Matrix m2 = new Matrix(2, 3);
        for(int i=0; i<2; i++) for(int j=0; j<3; j++) m2.array[i][j] = k++;
        m2.show();
        m4 = m3.transpose();
        System.out.println("전치행렬");
        m4.show();
        m5.multiply(m2);
        m5.show();
        m6 = m4.multiply(m5);
        m6.show();
        m7 = m6.multiply(m5);
        m7.show();
    }
}

Weeds are the blight of fields. Delusion is the blight of mankind.
Consequently offerings to those free from delusion are of great fruit.
358
zezeon@ubuntuZ:~$ cd java
/home/zezeon/Programming/java
zezeon@ubuntuZ:~/Programming/java$ make tex
//2016110056 박승원;
import java.io.*;
import java.util.*;
public class problem5 {
    static String trim(String s, int pos) { //return the part pos is located
        String r = "";
        for(int i=0; i<s.length(); i++) {
            if(s.charAt(i) == ' ') {
                if(i<pos) r = "";
            } else return r;
        }
        return r;
    }
}

----- 문제 problem5 번 실행을 시작합니다. -----
java problem5 w = JavaUtil.getInt();
찾을 파일명을 입력하세요 input.txt
찾을 문자열을 입력하세요 (comma로 구문을 띄우세요) :a,cd,A
a는 3개 : abcdefg aaA aaA
cd는 1개 : abcdefg
A는 2개 : aaA A
----- 문제 problem5 번 실행을 종료합니다. -----

"problem4.java" 94L, 2575C 저장 했습니다
90,16-25
바닥
```