

EXAMINATION
Ans-Final

ตัวอย่างโจทย์และแนวข้อสอบปลายภาค (2/2550)
(An Example of Final Examination Semester 2/2550)

โจทย์ข้อที่ 1 จงหาผลลัพธ์จากการทำงานของโปรแกรมต่อไปนี้ (10 คะแนน)

```
public class B extends A{
    A aInB;
    C cInB;
    int c;
    public B(){
        super();
        aInB = new A();
        cInB = new C();
    }
    public void print(){
        System.out.println("<<B>>");
    }
    public void print1(){
        print();
        c++;
    }
    public void print2(){
        super.print();
    }
    public void innerPrint(){
        aInB.print();
        aInB = new B();
        aInB.print();
        cInB.print();
        cInB.countPrint();
    }
    public void countPrint(String m){
        System.out.println(m + c);
    }
    public static void main(String [] s){
        B objB = new B();
        C objC = new C(5);
        /* =====Part 1===== */
        objB.print1();
        objB.print2();
        objB.print1();
        objB.countPrint();
        System.out.println(objB.c);
        objB.aInB.print();
        System.out.println(objB.aInB.c);
        objC.print();
        objC.countPrint();
        objB.innerPrint();
        /* =====Part 2===== */
        objB.aInB.print();
        System.out.println(objB.aInB.c);
        ((B) objB.aInB).aInB.print();
        System.out.println(objC.aInC.c);
        objC.aInC.print();
        objC.aInC.countPrint();
        objC.print();
        System.out.print(objC.aInC.c + ++objC.c);
        objB.countPrint("\nc = ");
    }
}
```

```
public class A{
    int c;
    public A(){
        c = 0;
    }
    public void print(){
        System.out.println("<<A>>");
        c++;
    }
    public void countPrint(){
        System.out.println(c);
    }
}
```

```
public class C{
    A aInC;
    int c;
    public C(){
        c = 10;
    }
    public C(int c){
        this.c = c;
        aInC = new A();
    }
    public void print(){
        System.out.println("<<C>>");
        c--;
    }
    public void countPrint(){
        System.out.println(c);
    }
}
```

ผลลัพธ์ Part 1

```
<<B>>
<<A>>
<<B>>
1
2
<<A>>
1
<<C>>
4
<<A>>
<<B>>
<<C>>
9
```

ผลลัพธ์ Part 2

```
<<B>>
0
<<A>>
0
<<A>>
1
<<C>>
5
c = 2
```

โจทย์ข้อที่ 2 ที่จอดรถหนึ่งจัดแบ่งพื้นที่จอดเป็นแบบ 2 มิติ ขนาด 20x40 สามารถจอดรถได้ทั้งหมด 800 คัน การระบุตำแหน่งของช่องจอดรถให้ใช้วิธีกำหนดตำแหน่ง Row และ Column แล้วให้เขียนโปรแกรมที่สมบูรณ์ในภาษาจาวาเพื่อทำงานดังนี้ (20 คะแนน)

1. จงเขียน Class ชื่อ CarPark และมี Object Method ดังต่อไปนี้
 - `getSpaces()` คืนค่าจำนวนช่องจอดรถว่างทั้งหมด
 - `getAddress()` คืนตำแหน่ง Row และ Column ของช่องจอดรถที่ยังว่างมา 1 ช่อง
 - `full()` ถ้าช่องจอดรถเต็มให้คืนค่า true แต่ถ้าช่องจอดรถว่างให้คืนค่า false
 - `park()` นำรถเข้าจอด โดยระบุตำแหน่ง Row และ Column ของช่องที่ว่างที่ต้องการจอด
 - `collect()` นำรถออกจากช่องจอดรถ โดยการระบุตำแหน่ง Row และ Column ของช่องที่จอดและทำให้ตำแหน่งนั้นเป็นที่ว่าง
2. จงเขียนโปรแกรมใน Main Class โดยการเรียกใช้ Object Method ของ CarPark เพื่อบริหารจัดการการจอดรถในที่จอด โดยเมื่อเริ่มโปรแกรมที่ช่องจอดรถทั้งหมดจะเป็นที่ว่างและแสดงข้อความบนจอภาพเพื่อรับข้อมูลจากแป้นพิมพ์ดังนี้

```
Enter Code (1-4) :
```

- ถ้าพิมพ์ 1 ตรวจสอบช่องว่าง ถ้ามีช่องว่างให้แสดงข้อความว่า ENTRY ถ้าไม่มีช่องว่างให้แสดงข้อความว่า NO ENTRY
- ถ้าพิมพ์ 2 เมื่อต้องการนำรถเข้าจอด โดยจะแสดงตำแหน่งช่องจอดรถที่ว่างขึ้นมา 1 ช่อง จากนั้นให้นำรถเข้าไปจอดในตำแหน่งช่องว่างนั้น
- ถ้าพิมพ์ 3 เมื่อเจ้าของรถต้องการนำรถออกจากที่จอด ก็จะต้องทำให้ช่องจอดรถนั้นเป็นที่ว่าง
- ถ้าพิมพ์ 4 ออกจากการทำงานของโปรแกรม

ตัวอย่างผลการทำงานของโปรแกรม

```
Enter Code (1-4) : 1
ENTRY
Enter Code (1-4) : 2
Row:0, Column:0
Car park ready
Enter Code (1-4) : 2
Row:0, Column:1
Car park ready
Enter Code (1-4) : 3
Row : 1
Column : 6
Car park not valid position
Enter Code (1-4) : 3
Row : 0
Column : 0
Car leave
Enter Code (1-4) : 2
Row:0, Column:0
Car park ready
Enter Code (1-4) : 4
Exit program
```

```
//Class CarPark
```

```
import jlab.JLabIO;
public class CarPark {
    private boolean[][] space; //false are default = ready for entry

    public CarPark() {
        space = new boolean[20][40];
    }

    public int getSpaces() {
        int countSpace = 0;
        for (int i = 0; i < space.length; i++) {
            for (int j = 0; j < space[i].length; j++) {
                if (!space[i][j]) countSpace++;
            }
        }
        return countSpace;
    }

    public int[] getAddress() {
        int[] cell = new int[2];
        for (int i = 0; i < space.length; i++) {
            for (int j = 0; j < space[i].length; j++) {
                if (!space[i][j]) {
                    cell[0] = i;
                    cell[1] = j;
                    return cell;
                }
            }
        }
        return cell;
    }

    public boolean full() {
        for (int i = 0; i < space.length; i++) {
            for (int j = 0; j < space[i].length; j++) {
                if (!space[i][j]) return false;
            }
        }
        return true;
    }

    public int park(int row, int column) {
        if (space[row][column]) return - 1;
        space[row][column] = true;
        return 0;
    }

    public int collect(int row, int column) {
        if (!space[row][column]) return - 1;
        space[row][column] = false;
        return 0;
    }
}
```

```
//Class Main
```

```
import jlab.JLabIO;
public class Main {
    public static void main(String[] args) {
        CarPark myCarPark = new CarPark();
        do {
            int code = JLabIO.readInt("Enter Code (1-4) : ");
            switch (code) {
                case 1 : {
                    if (myCarPark.full())
                        System.out.println("NO ENTRY");
                    else
                        System.out.println("ENTRY");
                    break;
                }
                case 2 : {
                    int[] space = myCarPark.getAddress();
                    System.out.println("Row:" + space[0] + ", Column:" + space[1]);
                    int park = myCarPark.park(space[0], space[1]);
                    if (park == 0) System.out.println("Car park ready");
                    else System.out.println("Not ready for park");
                    break;
                }
                case 3 : {
                    int row = JLabIO.readInt("Row : ");
                    int col = JLabIO.readInt("Column : ");
                    int collect = myCarPark.collect(row, col);
                    if (collect == 0) System.out.println("Car leave");
                    else System.out.println("Car park not valid position");
                    break;
                }
                case 4 : {
                    System.out.println("Exit program");
                    return;
                }
            } //End of switch
        } while (true);
    } //End of method main
} //End of class Main
```

โจทย์ข้อที่ 3 จงเขียนโปรแกรมภาษาจาวาให้สมบูรณ์ เพื่อค้นหาจำนวนคำตอบที่เป็นไปได้ทั้งหมดของสมการ $w^2+x^2+y^2+z^2 = 5000$ โดยที่ตัวแปร w, x, y และ z คือจำนวนเต็มใดๆ ที่ไม่ใช่จำนวนเต็มลบ และ $w \leq x \leq y \leq z$ (8 คะแนน)

```
public class SolutionCount {
    public static void main(String[] args) {
        int numSolutions = 0;
```

```

for(int w=0; w<=Math.sqrt(5000); w++) {
  for(int x=w; x<=Math.sqrt(5000); x++) {
    for(int y=x; y<=Math.sqrt(5000); y++) {
      for(int z=y; z<=Math.sqrt(5000); z++) {
        if(w*w + x*x + y*y + z*z == 5000) {
          numSolutions++;
        } //End of if
      } //End of for z
    } //End of for y
  } //End of for x
} //End of for w

```

```

System.out.println("The number of solutions is "+numSolutions);
}
}

```

โจทย์ข้อที่ 4 จงพิจารณาสมการต่อไปนี้เพื่อใช้สำหรับการเขียนโปรแกรมแบบ Recursive (6 คะแนน)

$$F(n) = \begin{cases} \sum_{i=1}^n (i^i + 2i) & n = 1, 2, \dots \\ 0 & \text{otherwise} \end{cases}$$



1. จงเปลี่ยนแปลงสมการ $F(n)$ ใหม่เพื่อให้อยู่ในรูปแบบของการเรียกใช้ตัวเอง (2 คะแนน)

$$F(n) = \begin{cases} F(n-1) + (n^n + 2n) & n = 1, 2, \dots \\ 0 & \text{otherwise} \end{cases}$$

2. จงสร้างเมทอด $f(n)$ ที่มีการทำงานแบบ Recursive เพื่อใช้คำนวณหาคำตอบของสมการ (4 คะแนน)

```

public int f(int n) {
  if (n < 1) return 0;
  else return f(n - 1) + Math.pow(n,n) + (2 * n);
}

```

โจทย์ข้อที่ 5 จงเขียนโปรแกรมที่มีการทำงานแบบ Recursive โดยโปรแกรมดังกล่าวจะต้องทำงานและให้ผลลัพธ์เหมือนกับโปรแกรมดังต่อไปนี้ (4 คะแนน)

```
public static double c(int n){
    double a=1, b=1, c=0;
    if(n<0) return c;
    if(n<=1) return 1;
    for(int i=2;i<=n;i++){
        c = b + 0.5*a*(i-1)/i-1;
        a = b;
        b = c;
    }
    return c;
}
```



```
public static double c(int n){
    if(n < 0) return 0;
    if(n <= 1) return 1;
    return c(n-1)+0.5*c(n-2)*(n-1)/n-1;
}
```

โจทย์ข้อที่ 6 จงแสดงผลการทำงานของการทำงานในการค้นข้อมูล 13 (key = 13) จากชุดข้อมูล num[] ที่มีการเรียงลำดับเรียบร้อยแล้ว โดยใช้อัลกอริทึมการค้นข้อมูลแบบ Binary Search ต่อไปนี้ (6 คะแนน)

```
public static int binarySearch(int[] sorted, int first, int last, int key) {
    while (first < last) {
        int mid = (first + last) / 2; //Compute mid point.
        if (key < sorted[mid]) {
            last = mid; //Repeat search in bottom half.
        } else if (key > sorted[mid]) {
            first = mid + 1; //Repeat search in top half.
        } else {
            return mid; //Found it. return position
        }
    }
    return -1; //Failed to find key
}

public static void main(String[] args) {
    int num [] = {0,1,3,3,4,5,7,8,8,8,9,10,11,13,15,15,17,19,20,22};
    int i = binarySearch(num, 0, num.length - 1, 13);
    System.out.println(i);
}
```

ผลลัพธ์จากการทำงาน

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	3	3	4	5	7	8	8	8	9	10	11	13	15	15	17	19	20	22
first									mid										last
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	3	3	4	5	7	8	8	8	9	10	11	13	15	15	17	19	20	22
									first					mid					last
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	3	3	4	5	7	8	8	8	9	10	11	13	15	15	17	19	20	22
									first			mid		last					
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	3	3	4	5	7	8	8	8	9	10	11	13	15	15	17	19	20	22
													mid	last					
																			first

โจทย์ข้อที่ 7 จงแสดงผลการทำงานของการทำงานในการจัดเรียงข้อมูลคั่น data[] โดยใช้อัลกอริทึม Insertion Sort
ต่อไปนี้ (4 คะแนน)

```
public static void insertionSort(int[] list, int length) {
    int firstOutOfOrder, location, temp;
    for(firstOutOfOrder = 1; firstOutOfOrder < length; firstOutOfOrder++) {
        if(list[firstOutOfOrder] < list[firstOutOfOrder - 1]) {
            temp = list[firstOutOfOrder];
            location = firstOutOfOrder;
            do {
                list[location] = list[location-1];
                location--;
            } while (location > 0 && list[location-1] > temp);
            list[location] = temp;
        }
    }
}

public static void main(String[] args) {
    int data [] = {10,12,8,1,6,19,4,5,8,11};
    insertionSort(data, data.length);
}
```

ผลลัพธ์จากการทำงาน

เริ่มต้น	10	12	8	1	6	19	4	5	8	11
รอบที่ 1	10	12	8	1	6	19	4	5	8	11
รอบที่ 2	8	10	12	1	6	19	4	5	8	11
รอบที่ 3	1	8	10	12	6	19	4	5	8	11
รอบที่ 4	1	6	8	10	12	19	4	5	8	11
รอบที่ 5	1	6	8	10	12	19	4	5	8	11
รอบที่ 6	1	4	6	8	10	12	19	5	8	11
รอบที่ 7	1	4	5	6	8	10	12	19	8	11
รอบที่ 8	1	4	5	6	8	8	10	12	19	11
รอบที่ 9	1	4	5	6	8	8	10	11	12	19

โจทย์ข้อที่ 8 จงสร้าง Class Box ที่เป็นตัวแทนของคลาสกล่องซึ่งประกอบไปด้วยรายละเอียดต่างๆ
ดังต่อไปนี้ (10 คะแนน)

- 1) แอตทริบิวต์ 3 ตัว คือ double width, double height และ double length ในการเก็บค่าความกว้าง ความยาว และความสูง ของกล่องตามลำดับ
- 2) Constructor 2 ตัว เพื่อใช้กำหนดรูปทรงของกล่องว่าเป็นทรงสี่เหลี่ยมผืนผ้า หรือทรงสี่เหลี่ยมจัตุรัส
- 3) เมธอด 5 เมธอด ประกอบไปด้วยเมธอด area() ใช้หาค่าพื้นที่ผิวของกล่อง เมธอด volume() ใช้หาปริมาตรของกล่อง เมธอด length() เมธอด height() และเมธอด width() ใช้คืนค่าความกว้าง ความยาว และความสูง ของกล่องตามลำดับ

```
class Box {
    private double width;
    private double height;
    private double length;

    Box(double width1,double height1,double length1) {
        width = width1;
        height = height1;
        length = length1;
    }

    Box(double side) {
        width = side;
        height = side;
        length = side;
    }

    public double volume() {
        double volumeBox;
        volumeBox = width * height * length;
        return volumeBox;
    }

    public double area() {
        double areaBox;
        areaBox = 2*((width*length)+(width*height)+(length*height));
        return areaBox;
    }

    public double length() {
        return length;
    }

    public double height() {
        return height;
    }

    public double width() {
        return width;
    }
}
```

โจทย์ข้อที่ 9 Given two class definitions for A and B below. (5 Points)

```
public class A {
    public int x;
    public static int y;
    public A() {
        System.out.println("A's constructor");
        x = 10;
        y = 5;
    }
    public void f() { System.out.println(x-y); }
    public static void g() { System.out.println(y); }
    public String toString() { return "A: " + x + ", " + y; }
}
```

```
public class B extends A {
    public int x;
    public static int z;
    public B() {
        System.out.println("B's constructor");
        x = 7;
        z = 2;
    }
    public void f() { System.out.println(x+y+z); }
    public static void g(int x) { System.out.println(z); }
    public String toString() {
        return super.toString() + " B: " + x + ", " + z;
    }
}
```

What would be the output of each of the following code segments? Assume that they are executed in sequence from top to bottom.

A a = new A();	output A's constructor
B b = new B();	output A's constructor B's constructor
System.out.println(a);	output A: 10, 5
System.out.println(b);	output A: 10, 5 B: 7, 2
A x = a; x.f();	output 5
x.g();	output 5
x = b; x.f();	output 14
x.g();	output 5

โจทย์ข้อที่ 10 จงเขียนโปรแกรมภาษาจาวาเพื่อคำนวณหาคำตอบของสมการ $A^n + nA$ โดยที่ A เป็นเมตริกที่มีขนาด $m \times m$ (จำนวนสมาชิกของแถวและหลักเท่ากัน) ซึ่งจะต้องใช้หลักการของ Method และ Class หรือ Object ได้อย่างอิสระ แต่จะต้องมีเมทอดบังคับคือ เมทอด addMatrix(...) ใช้เพื่อคำนวณการบวกกันของสองเมตริก และเมทอด mulMatrix(...) ใช้เพื่อคำนวณการคูณกันของสองเมตริก (15 คะแนน)

```
import jlab.JLabIO;
public class Matrix {
    public static int[][] addMatrix(int a[][], int b[][]) {
        int c[][] = new int[a.length][a[0].length];
        for (int i = 0; i < a.length; i++)
            for (int j = 0; j < a[i].length; j++)
                c[i][j] = a[i][j] + b[i][j];
        return c;
    }
}
```

```
public static int[][] addMetrix(int a[][], int n) {
    int c[][] = new int[a.length][a[0].length];
    c = addMetrix(a, a);
    for (int i = 2; i < n; i++) {
        c = addMetrix(c, a);
    }
    return c;
}

public static int[][] mulMetrix(int a[][], int b[][]) {
    int[][] c = new int[a.length][b[0].length];
    for (int i = 0; i < a.length; i++) {
        for (int j = 0; j < b[i].length; j++) {
            c[i][j] = 0;
            for (int k = 0; k < b.length; k++) {
                c[i][j] += a[i][k] * b[k][j];
            }
        }
    }
    return c;
}

public static int[][] mulMetrix(int a[][], int n) {
    int[][] c = new int[a.length][a[0].length];
    c = mulMetrix(a,a);
    for (int i = 2; i < n; i++) {
        c = mulMetrix(c,a);
    }
    return c;
}

public static void printMetrix(int[][] a) {
    for (int i = 0; i < a.length; i++) {
        System.out.print("| ");
        for (int j = 0; j < a[0].length; j++) {
            System.out.print(a[i][j] + " ");
        }
        System.out.println("|");
    }
}

public static void main(String[] args) {
    int m = JLabIO.readInt("Enter Matrix Size (m x m) : ");
    int n;
    do {
        n = JLabIO.readInt("Enter n : ");
    } while (n < 2);

    int a[][] = new int[m][m];
    for (int i = 0; i < m; i++)
        for (int j = 0; j < m; j++)
            a[i][j] = JLabIO.readInt("Enter a[" + i + "][" + j + "] : ");
    printMetrix(addMetrix(mulMetrix(a,n),addMetrix(a,n)));
}
}
```