단국대학교 응용컴퓨터공학 JAVA 프로그래밍 II 중간고사 (2017년 봄학기) 2017년 4월 26일

학과 _____ 학번 _____ 이름 _____

중간고사

담당교수: 단국대학교 응용컴퓨터공학 박경신

- **답은 반드시 답안지에 기술할 것. 공간이 부족할 경우 반드시 답안지 몇 쪽의 뒤에 있다고 명기한 후 기술할 것. 그 외의 경우의 답안지 뒤쪽이나 연습지에 기술한 내용은 답안으로 인정 안 함. 답에는 반드시 네모를 쳐서 확실히 표시할 것.**
- **답안지에 학과, 학번, 이름 외에 본인의 암호(4자리 숫자)를 기입하면 성적공고시 학번대신 암호를 사용할 것임.**

1. Value 클래스이다. (1.1) ~ (1.8)에 주석(코드설명)을 달아라. (10점)

```
import java.util.Comparator;
import java.util.Objects;

public class Value implements Comparable<Value> {

	private int a;
	private int b;

	public Value() { // _(1.1)__Value 생성자_____________________________________
		this(0, 0);
	}

	public Value(int a, int b) {
		set(a, b);
	}

	public Value(int[] ab) {
		set(ab[0], ab[1]);
	}

	public void set(int a, int b) { // __(1.2)_a,b 를  지정하는 setter_____________
		this.a = a;
		this.b = b;
	}

	public int getA() { // __(1.3)_a 를 반환하는 getter_____________________________
		return this.a;
	}

	public int getB() { // __(1.3) _b 를 반환하는 getter_____________________________
		return this.b;
	}

	@Override
	public String toString() { // __(1.4)_Object 클래스의 toString() 메소드 재정의__
		return this.a + "," + this.b;
	}

	@Override
```

학과 _____ 학번 _____ 이름 _____

```
	public boolean equals(Object other) { // _(1.5) Object equals 메소드 재정의___
		if (this == other) return true;
		if (other instanceof Value) { // _(1.6) other 객체가 Value 객체인지__
			Value that = (Value)other;
			return Objects.equals(this.a, that.a)
				&& Objects.equals(this.b, that.b);
		}
		return false;
	}

	@Override
	public int hashCode() { // __(1.7) Object hashCode 메소드 재정의_______
		return Objects.hash(this.a, this.b); // hash 값 생성
	}

	@Override
	public int compareTo(Value other) { // _(1.8) Comparable compareTo 메소드 재정의_
		return this.a - other.a; // a 로 비교
	}

	// B 를 비교하는 Comparator<Value> 클래스 구현
	public static Comparator<Value> BComparator = new Comparator<Value>() {
		public int compare(Value v1, Value v2) { //Comparator compare 메소드재정의
			return v1.getB() - v2.getB(); // b 로 비교
		}
	};
}
```

2. Arithmetic 클래스의 min, max, round, pow 메소드 내부 코드를 완성하라 (20점)

```
public class Arithmetic
{
	public static int add(int n, int m) {
		return n + m;
	}

	public static int sub(int n, int m) {
		return n - m;
	}

	public static int multiply(int n, int m) {
		return n * m;
	}

	public static int divide(int n, int m) throws ArithmeticException {
		return n / m;
	}

	public static int abs(int n) { // abs(-5)=5 abs(3)=3
		return n > 0 ? n : -n;
	}

	public static int min(int n, int m) { // min(5,10)=5 min(-5,-10)=-10
		return _n < m ? n : m;_____________________________
	}
```

학과 _____ 학번 _____ 이름 _____

```
        public static int max(int n, int m) { // max(5,10)=10 max(-5,-10)=-5
                return _n > m ? n : m;_____________________________
        }

        public static int round(double n) { // round(5.1)=5 round(5.7)=6 round(-5.8)=-6
                return _n > 0 ? (int) (n + 0.5) : (int) (n -0.5);___
        }

        public static int pow(int n, int a) { // pow(2,5)=2*2*2*2*2=32
                if (a == 0) return 1;
                return _n * pow(n , a - 1);_____________________
        }
}
```

3. Lab 클래스이다. 빈 칸을 채워라 (20점)

```
public class Lab {

        _static int input1, input2;_____________________ // input1, input2

        _static Scanner scan = new Scanner(System.in);____ // Scanner 객체 생성
        public static void getUserInput() {
                System.out.print("Please enter [input1]: ");
                input1 = scan.nextInt();
                System.out.print("Please enter [input2]: ");
                input2 = scan.nextInt();
                scan.nextLine();
        }

        public static ___Value[]____ getFileInput(__String filename__) {
                Value[] vArray = new Value[7];
                try {
                        BufferedReader br = new BufferedReader(new FileReader(filename));
                        String line = "";
                        int index = 0;
                        while (_(line = br.readLine()) != null_) { // read line until EOF
                                String[] values = line.split(","); // line -> values[]
                                int a = Integer.parseInt(values[0]); // String -> int
                                int b = Integer.parseInt(values[1]); // String -> int
                                __vArray[index] = new Value(a, b);_ // add Value to vArray
                                index++;
                        }
                        br.close();
                } catch (IOException e) {
                        e.printStackTrace();
                } catch (NullPointerException e) {
                        e.printStackTrace();
                }
                return ____vArray;_______
        }

        public static void print(Value[] values) {
                for (___Value v : values___) { // foreach
                        System.out.println(v);
                }
        }
```

학과 _____ 학번 _____ 이름 _____

4. Lab 클래스의 main() 메소드에서 각 질문에 답하라 (50점)

"valuelist.ini"
1,2
3,4
5,6
7,8
9,10
11,12
13,14

```
public static void main(String[] args) {

        // FileIO
        Value[] vArray = getFileInput("valuelist.ini");
        print(vArray);

        // (3.1) (5점)
        int[] val1 = {-3, 4};
        vArray = new Value[5];
        vArray[0] = new Value(1, 2);
        vArray[1] = new Value(val1[0], val1[1]);
        vArray[2] = new Value(val1);
        vArray[3] = new Value(6, -5);
        vArray[3] = new Value(-7, -8);
        vArray[4] = vArray[3];
        print(vArray);

        // (3.2) (5점)
        if (vArray[1] == vArray[2])
                System.out.println("vArray[1] == vArray[2]");
        if (vArray[1].equals(vArray[2]))
                System.out.println("vArray[1] equals vArray[2]");
        if (vArray[3] == vArray[4])
                System.out.println("vArray[3] == vArray[4]");
        if (vArray[3].equals(vArray[4]))
                System.out.println("vArray[3] equals vArray[4]");

        // (3.3) (10점)
        vArray = new Value[2];
        Value value1 = new Value();
        for (int i=0; i<2; i++) {
                getUserInput();
                value1.set(input1, input2);
                vArray[i] = value1;
        }
        print(vArray);
        for (int i=0; i<2; i++) {
                getUserInput();
                Value value2 = new Value();
                value2.set(input1, input2);
                vArray[i] = value2;
        }
        print(vArray);

        // (3.4) (5점)
        List<Value> vList = new ArrayList<Value>();
        vList.add(new Value(1, 2));
        vList.add(new Value(1, 2));
        vList.add(new Value(-3, 4));
        vList.add(new Value(-3, 4));
        vList.add(new Value(6, -5));
        vList.add(new Value(-7, -8));
        vList.add(new Value(-7, -8));
        vList.forEach((v) -> System.out.println(v));
```

학과 _____ 학번 _____ 이름 _____

```
        // (3.5) (10점)
        vList.sort((Value v1, Value v2)-> v1.getA() - v2.getA());
        vList.forEach((v) -> System.out.println(v));
        vList.sort(Value.BComparator);
        vList.forEach((v) -> System.out.println(v));
        vList.sort(null);
        vList.forEach((v) -> System.out.println(v));

        // (3.6) (5점)
        List<Value> fList = vList.stream()
                                 .filter(p -> p.getA() < 0 && p.getB() < 0)
                                 .collect(Collectors.toList());
        fList.forEach(p -> System.out.println(p));

        // (3.7) (10점)
        Set<Value> vSet = new HashSet<Value>();
        vSet.add(new Value(1, 2));
        vSet.add(new Value(1, 2));
        vSet.add(new Value(-3, 4));
        vSet.add(new Value(-3, 4));
        vSet.add(new Value(6, -5));
        vSet.add(new Value(-7, -8));
        vSet.add(new Value(-7, -8));
        vSet.forEach((v) -> System.out.println(v));
    }
}
```

3.1 main()에 (3.1) 실행결과를 적고 그 이유를 설명하라. (5점)

```
1, 2 // Value(int a, int b) 사용
-3,4 // Value(int a, int b) 사용
-3,4 // Value(int[] ab) 사용
-7,-8 // vArray[3]은 마지막 것으로 치환
-7,-8 // vArray[4]는 vArray[3]을 대입
```

3.2 main()에 (3.2) 실행결과를 적고 그 이유를 설명하라. (5점)

```
vArray[1] equals vArray[2] // vArray[1]과 vArray[2]는 내용은 같으나 같은 레퍼런스를 가리키지 않음
vArray[3] == vArray[4] // vArray[1]과 vArray[2]는 같은 레퍼런스이며,
vArray[3] equals vArray[4] // vArray[1]과 vArray[2]는 내용도 같음
```

3.3 main()에 (3.3) 실행결과를 적고 그 이유를 설명하라. (10점)

```
Please enter [input 1] : 1
Please enter [input 2] : 2
Please enter [input 1] : 3 // value1.set(3,4); value1은 한 개만 생성되어서 공유해서 사용
Please enter [input 2] : 4 // value1.set(3,4); 마지막 set의 값만 가짐
3,4
3,4
Please enter [input 1] : 1 // value2는 for 내부에서 각각 생성되어 따로 set하므로
Please enter [input 2] : 2
Please enter [input 1] : 3
Please enter [input 2] : 4
1,2
3,4
```

학과 _____ 학번 _____ 이름 _____

3.4 main()에 (3.4) 실행결과를 적고 그 이유를 설명하라. (5점)

```
1, 2 // ArrayList에 Value(1, 2)를 추가
1, 2 // ArrayList에 Value(1, 2)를 추가
-3,4 // ArrayList에 Value(-3, 4)를 추가
-3,4 // ArrayList에 Value(-3, 4)를 추가
6,-5 // ArrayList에 Value(6, -5)를 추가
-7,-8 // ArrayList에 Value(-7, -8)를 추가
-7,-8 // ArrayList에 Value(-7, -8)를 추가
```

3.5 main()에 (3.5) 실행결과를 적고 sort 방식의 차이점을 자세히 설명하라. (10점)

```
-7,-8 // vList.sort((Value v1, Value v2) -> v1.getA() - v2.getA()); // (1) 람다식으로 A로 정렬
-7,-8
-3,4
1,2
1,2
6,-5
-7,-8 // vList.sort(Value.BComparator); // (2) BComparator 클래스를 이용한 B로 정렬
-7,-8
6,-5
1,2
1,2
-3,4
-7,-8 // vList.sort(null); // (3) Comparable<Value> compareTo를 이용한 A로 정렬
-7,-8
-3,4
1,2
1,2
6,-5
```

3.6 main()에 (3.6) 실행결과를 적고 그 이유를 설명하라. (5점)

```
fList의 결과는 A,B가 모두 음수인 것을 모두 필터해서 새로운 리스트로 반환한다. 따라서
-7,-8
-7,-8
```

3.7 main()에 (3.7) 실행결과를 적고 그 이유를 설명하라. (10점)

```
HashSet은 동일한 key값을 허용하지 않는다. Value 클래스에 hashCode()가 재정의되어 있기 때문에
같은 내용을 갖는 Value는 동일한 hashCode를 가지므로 중복될 수 없다. 따라서
-7,-8
1,2
6,-5
-3,4
```

-끝-