



United International University (UIU)
Dept. of Computer Science and Engineering (CSE)

MID EXAM :: SUMMER 2019

Course Code: **CSI 211** Course Title: **Object-Oriented Programming**

Total Marks: **30** Time: **1 hour 45 minutes**

1. a) A complex number consists of a real and imaginary number. Consider the following Main class and the Output section. You have to write a class named "Complex" to generate the desired output.

3+3

```
public class Main {

    public static void main(String[] args) {
        Complex c1 = new Complex(10, 15);
        Complex c2 = new Complex(c1);

        System.out.println(c1.real+" "+c1.img);
        System.out.println(c2.real+" "+c2.img);
    }

}
```

Output:

10 15

10 15

- b) What will be the output of the following code.

```
class A {
    int a;
    int b;
    public A(int a) {
        this.a = a;
    }
    public A(int a, int b) {
        this.a = a;
        this.b = b;
    }
    public int subtract() {
        return a - b;
    }
}
```

```
class B extends A {
    public B(int a, int b)
    {
        super(a);
    }
    public int subtract()
    {
        if (a >= b) {
            return a - b;
        } else {
            return b - a;
        }
    }
}
```

```
public class FinalQues {
    public static void main(String[] args)
    {
        A obj = new B(-5, 5);

        System.out.println(obj.subtract());
    }
}
```

2. a) Suppose, one has written the following code to print the name of Student s1. Is there any error there? If so, you have to modify the code so that it would print the student s1's name. You can add extra methods to **Student** class but cannot change any existing lines there. You can modify existing lines in **main** method of **Test** class.

3+3

```
Package pack;

public class Student
{
    private String name;

    Student(String n)
    {
        name=n;
    }
}
```

```
Package pack;

Public class Test{
    public static void main(String[] args)
    {
        Student s1=new Student("Sam");
        System.out.println(s1.name);
    }
}
```

- b) Is there any error in the code given in the next page. If so, you can only modify the main method to correct the code. You can only edit, you **cannot** delete or add any new line while modifying. Write the outputs of the code after modification.

<pre> Package pack; public class Shape { void introduce() { System.out.println ("It is shape class"); } } </pre>	
<pre> Package pack; public class Triangle extends Shape { @Override void introduce() { System.out.println ("It is Triangle class"); } void calcArea() { System.out.println ("Calculate Area"); } } </pre>	<pre> Package pack; Public class Test{ public static void main(String[] args) { Shape s = new Triangle(); s.introduce(); S.calcArea(); } } </pre>

3. a) Write a code fragment to create the following multidimensional integer array.

1	2	3	4
5	6	7	
8	9		
10			

b) Write a Java program that will go through the items of **an array** and find the sum of the value of the array using **enhance for loop**. Take the following values as the initial values of the array {1, 3, 4, 8, 2, 7, 5, 9, 6}

c) How many objects are eligible for garbage collection after executing the following code (show graphically)?

```

public class Student {
    String name;
    Student(String n)
    {
        name=n;
    }
    public static void main(String[] args)
    {
        Student s1=new Student("Sam");
        Student s2=new Student("Mike");
        Student s3=new Student("John");
        s1=s2;
        s1=s3;
    }
}

```

2+2+2

4. a)What will be the output of the following program?

1.5+
1.5+3

```
class Student {
    public String name;
    public int id;
    Student(String name,int id){
        this.name = name;
        this.id = id;
    }

    public Student getStudent(Student s1){
        s1 = new Student("Rahim",101);
        return s1;
    }
}
class Main{
    public static void main(String[] args) {
        Student s1 = new Student("Karim",102);
        System.out.println("Name : " + s1.name + " ID : " + s1.id);
        Student s2 = s1.getStudent(s1);
        Student s3 = s2;
        System.out.println("Name : " + s3.name + " ID : " + s3.id);
    }
}
```

b) Find out the errors in Table-1 and Table-2 and correct them. If you do not find any error, simply write “No error” for each code fragment.

Table – 1

```
class B{
    void print(){
        System.out.println("this is B
class...");
    }
}
public class A {
    public static void main(String[] args) {
        B B_obj = new B();
    }
}
```

Table-2

```
class B{
    B(int a){
        System.out.println(a);
    }
    void print(){
        System.out.println("this is B class..");
    }
}
public class A {
    public static void main(String[] args) {
        B B_obj = new B();
    }
}
```

c) Write a class named Box which has 3 instance variables[**width(double)** , **height(double)** and **depth(double)**] ,one instance method[**double volume()**] and **one constructor**. Create two classes named **Cube** and **DifferentShapedBox** by inheriting the Box class. Is it possible to write the above program using “Abstract classes” ? If possible then:

- i) Complete the program using abstract class
- ii) What is the advantage of using abstract class in question no – i?

5. a) Is the following program correct? If yes, write the output of the following program. If not, please explain why it will not be working.

3+3

```
class Base {
    Base(){
        System.out.println("Base called");
    }
    public void show() {
        System.out.println("Base::show() called");
    }
}

class Derived extends Base {
    Derived(){
        System.out.println("Derived called");
    }
    public void show() {
        System.out.println("Derived::show() called");
    }
}

public class Main {
    public static void main(String[] args) {
        Base b = new Derived();
        b.show();
    }
}
```

- b) What will be the output of the following program?

```
class Base {
    public void Print() {
        System.out.println("Base");
    }
}

class Derived extends Base {
    public void Print() {
        System.out.println("Derived");
    }
}

class Main{
    public static void DoPrint( Base o ) {
        o.Print();
    }
    public static void main(String[] args) {
        Base x = new Base();
        Base y = new Derived();
        Derived z = new Derived();
        DoPrint(x);
        DoPrint(y);
        DoPrint(z);
    }
}
```