Ahmadu Bello University, Zaria
Department of Computer Science

2016/2017 First Semester Examination
COSC 211 Object Oriented Programming I

Date: March xx, 2017 **Time Allowed**: 120 Minutes

Instructions: Attempt **ANY FOUR** questions.

1. (**20 marks**). Trace the output of each of the following programs. Write your answers in the spaces provided.
   a. (4 marks).

```
public class ExamQ1A {
   public static void main(String[] args)  {
          int ctemp = 73, ftemp;
          ftemp = 9*ctemp/5 + 32;
          System.out.println("ftemp = " + ftemp);
   }
}
```

Answer:

```
ftemp = 163
```

   b. (4 marks). Assume the user entered: **Muhammadu 19 u16cs12345**

```
import java.util.Scanner;
public class ExamQ1B {
  public static void main(String[] args)  {
          int x1;
          String s1, s2, password;
          Scanner keyboard = new Scanner(System.in);
          System.out.println("Enter your first name, age, ID#: ");
          s1 = keyboard.next();
          x1 = keyboard.nextInt();
          s2 = keyboard.next();
          password = s1.substring(0, 5) + (x1 % 5) + s2.charAt(3);
          System.out.println("Your password is: " + password);

  }
}
```

Answer:

```
Your password is: Muham4c
```

c. (6 marks). Assume the user entered: 3

```
import java.util.Scanner;
public class ExamQ1C {
  public static void main(String[] args)   {
          int c, q;
    Scanner keyboard = new Scanner(System.in);
    System.out.println("Enter an integer from 1 to 3: ");
    c = keyboard.nextInt();
    q = 4 - c;
    switch(q)    {
      case 1: System.out.println("Hi, welcome to java");
      case 2: System.out.println("c and q are equal!"); break;
      case 3: System.out.println("You typed 1");
      default: System.out.println("Goodbye"); break;
     }
  }
}
```

Answer:

```
Hi, welcome to java
c and q are equal!
```

d. (6 marks).

```
public class ExamQ1E {
  public static void main(String[] args)   {
          int i, sum = 0;    double avg;
    for(i = 5; i < 15; i++)    {
          if(i % 2 == 0) sum = sum + i;
          else sum = sum + 2;
    }
    avg = sum/10.0;
    System.out.println("The sum is: " + sum);
    System.out.println("The average is: " + avg);
  }
}
```

Answer:

```
The sum is: 60
The average is: 6.0
```

2. (**20 marks**). Answer the following questions about two-dimensional arrays.
   a. (**10 marks**). Write a method **public static void printRow(int i, double [][] myArray)**, that prints the i^{th} row of its two-dimensional array parameter, **myArray**.

```
public static void printRow(int i, double[][] myArray){
        for(int j = 0;j<myArray[i].length;j++)
            System.out.println(myArray[i][j]);
}
```

   b. (**10 marks**). Write Java code segment to illustrate how the method **printRow** above can be called. ***Hint***: you must define a two-dimensional array and pass it as parameter to the method.

```
public static void main(String [] args){
        double[][] values = {{2,4,5},{7,9,2},{8,6,4}};
        printRow(1,values);
}
```

3. (**20 marks**). Answer both the following questions
   a. (**10 marks**) Write a recursive method **public int recursive(int x, int y)** that multiplies two integers **x** and **y** using repeated additions and without using multiplication.

```
public int recursive(int x, int y){
        if(y == 1)
            return x;
        else return x + recursive(x, y-1);
}
```

   b. (**10 marks**) Write a iterative method **public int iterative(int x, int y)** that multiplies two integers **x** and **y** using repeated additions and without using multiplication.

```
public int iterative(int x, int y){
        int mult = 0;
        while(y > 0){
            mult +=x;
            y--;
        }
    return mult;
}
```

4. (**20 marks**). Answer both questions.
   a. (**10 marks**). Study the following program carefully and write down its output.

```
class Parent {
  static int x;
  public Parent(int x){
    f(x);
    g();
  }
```

```
  public void f(int x){
    System.out.println(x);
  }
  public static void g(){
    System.out.println(x);
  }
}
public class FinalQ4a extends Parent {
  static int x = 7;
  public FinalQ4a(int x){
    super(x);
    g();
    f(x);
  }
  public void f(int x){
    System.out.println(x*x);
  }
  public static void g(){
    System.out.println(x*x);
  }
  public static void main(String []s){
    Parent obj = new FinalQ4a(3);
    obj.g();
  }
}
```

Answer:

```
9
0
49
9
0
```

b. (**10 marks**). Study the following program carefully and write down its output.

```
public class FinalQ4b {
  public static void main(String[] args){
      int[] grades = {5, 7, 6, 8, 10};
      System.out.println("Grades before:");
      printArray(grades);
      doubleArray(grades);
      System.out.println("Grades after:");
```

```
    printArray(grades);
  }
  public static void printArray(int[] a){
    for(int i = a.length-1; i >= 0; i--)
      System.out.print(a[i]+ "   ");
    System.out.println();
   }
  public static void doubleArray(int[] a){
     for(int i = a.length-1; i >= 0; i--)
       a[i] = a[i] - a[0];
   }
}
```

Answer:

```
Grades before:
10   8   6   7   5
Grades after:
5   3   1   2   0
```

5. (**20 marks**). Implement a superclass Person. A person has a name and a year of birth. Make two classes, Student and Instructor, inherit from Person. A student has a major, and an instructor has a salary. Write the class definitions, the constructors, and the method **toString** for all classes. Supply a test program that tests these classes and methods.

```
// Person.java
class Person{
        private String name;
        private int year;
        public Person(String name,int year){
                this.name = name;
                this.year = year;
        }
        public String toString(){
                return "Name: " + name + "\nBirth Date: " + year;
        }
}

//Student.java
class Student extends Person{
        private String major;
        public Student(String name, int year, String major){
                super(name,year);
                this.major = major;
        }
        public String toString(){
                return super.toString()+"\nMajor: "+major;
```

```
        }
}

// Instructor.java
class Instructor extends Person{
        private double salary;
        public Instructor(String name,int year,double salary){
                super(name,year);
                this.salary = salary;
        }
        public String toString(){
                return super.toString() + "\nSalary: " + salary;
        }
}

// TestPerson.java
public class TestPerson{
        public static void main(String[] args){
                Person s = new Student("Uthman Aliyu",2011,"COSC211");
                Person i = new Instructor("Dr. Aliyu Salisu",1981,180000.00);
                System.out.println(s);
                System.out.println(i);
        }
}
```

6. Select the most appropriate matching answer for each of the following questions. Write all your answers in the following table. → 2 marks each

I	ii	iii	iv	v	vi	vii	viii	ix	x
C	**d**	**a**	**d**	**a**	**c**	**d**	**a**	**b**	**c**

i. Which of the following is not correct?
   a. Exceptions of class IOException and its subclasses should be handled.
   b. User-errors should be handled
   c. The exceptions of class Error and its subclasses should be handled
   d. None of the options is correct.

ii. Which of the following is not correct?
   a. A method that does not handle checked exceptions it may generate, must declare those exceptions in a throws clause; otherwise a compile-time error occurs.
   b. A method may or may not declare, in a throws clause, unchecked exceptions it may generate but does not handle.
   c. When a method declares that it throws an exception, then it can throw an exception of that class or any of its subclasses.
   d. When a method declares that it throws an exception, then it can throw an exception of that class or any of its superclasses.
   e. None the options is correct

iii. Consider the following code segment and select the most appropriate option.

```
try{
    int x,y = 0;
     x =x / y;
    }
    catch(Exception  e){
      System.out.println("First catch");
    }
    catch(ArithmeticException e){
      System.out.println("Second catch");
    }
```

   a. The code will cause compile-time error
   b. Will display: First catch
   c. Will display : Second catch
   d. None of the options is correct

iv. If you declare an array double[] list = {3.4, 2.0, 3.5, 5.5}, the highest index in array list is _____.
   a. 0
   b. 1
   c. 2
   d. 3

v. What is output of the following code:

```
public class Test {
  public static void main(String[] args) {
    int list[] = {1, 2, 3, 4, 5, 6};
```

```
    for (int i = 1; i < list.length; i++)
      list[i] = list[i - 1];

    for (int i = 0; i < list.length; i++)
      System.out.print(list[i] + " ");
  }
}
```

a. 1 2 3 4 5 6
b. 2 3 4 5 6 6
c. 2 3 4 5 6 1
d. 1 1 1 1 1 1

vi. Show the output of the following code:

```
public class Test {
  public static void main(String[] args) {
    int[] x = {1, 2, 3, 4, 5};
    increase(x);
    int[] y = {1, 2, 3, 4, 5};
    increase(y[0]);
    System.out.println(x[0] + " " + y[0]);
  }

  public static void increase(int[] x) {
    for (int i = 0; i < x.length; i++)
      x[i]++;
  }

  public static void increase(int y) {
    y++;
  }
}
```

a. 1 1
b. 2 2
c. 2 1
d. 1 2

vii. Suppose a method p has the following heading:

```
public static int[] p()
```

What return statement may be used in p()?

a. return 1;
b. return {1, 2, 3};

c. return int[]{1, 2, 3};
d. return new int[]{1, 2, 3};

viii. What exception type does the following program throw?

```
public class Test {
  public static void main(String[] args) {
    System.out.println(1 / 0);
  }
}
```

a. ArithmeticException
b. ArrayIndexOutOfBoundsException
c. StringIndexOutOfBoundsException
d. ClassCastException

ix. Object-oriented programming allows you to derive new classes from existing classes. This is called _____.

a. encapsulation
b. inheritance
c. abstraction
d. generalization

x. Which of the following classes cannot be extended?

a. class A { }
b. class A { private A();}
c. final class A { }
d. class A { protected A();}