BIRZEIT UNIVERSITY

Faculty of Engineering and Information Technology

Computer Science Department

Advance Programming -Comp231
Midterm Exam
1st Semester 2021/2022

Student Name: _____ Student ID: _____

Circle the your lecture and lab number :

Instructor Name	Lecture No.	LAB No. With Days			
Dr.Yousef Hassouneh	1	1	W (2:15-16:55)	2	T(11:25-2:05)
Mr.Hafez Barghouthi	2	3	M(2:15-16:55)	4	W(2:15-16:55)
Dr.Bassem Sayrafi	3	5	T(8:00-10:40)	6	R(2:15-16:55)
Mr.Nael Qaraeen	4	7	W(8:00-10:40)	8	T(11:25-2:05)
Dr.Anas Arram	5	9	R (11:25-2:05)	10	M(2:15-16:55)
Dr. Mohammad Alkhanafseh	6	11	M(2:15-16:55)	12	R (11:25-2:05)
Dr.Mamoun Nawahdah	7	13	M (8:00-10:40)	14	R(8:00-10:40)
Mr.Murad Njoum	8	15	W (8:00-10:40)	16	M (11:25-2:05)
Ms.Dima Taji	9	17	T(8:00-10:40)	18	R(2:15-16:55)

Question Number	Grade
Q#1	**/30**
Q#2	**/16**
Q#3	**/20**
Q#4	**/40**
Grade **(6 marks are bonus)**	

Instructions:

- Fill in your information above.
- Do NOT start the exam until you are instructed to do so.
- This is a closed material exam.
- Calculators are NOT allowed, if you have one, put it on the floor.
- Mobile phones are NOT allowed, if you have one, switch it off NOW.
- Smart watches are NOT allowed, if you have one, take it off NOW.
- **The questions will be graded from the answer form. So, transfer your answers to it.**

Question#1[30 Marks]Choice the best Answer:

[1] .Which of the following statements correctly create an array of five empty Strings?

```
1. String[] a=new String[5];
   for(int j=0;j<5;j++)
            a[j]="";
2. String[] a={"","","","",""};
3. String a[5];
4. String [5] a;
5. String [] a=new String[5];
    a[]={"","","","",""};
```

A. None B.Only 1 C.Only 1 and 2 D.Only 1,2 and 5

[2]. What is the value of n ?

```
String str="Comp231 midterm exam";
int n= str.indexOf("231",str.length()-1);
```

A. 0 B. -1 C. 4 D.Compiler error

[3]. Given the following code, what is the output ?

```
StringBuilder sb = new StringBuilder("Hi");
System.out.println(sb.length());
```

A. 0 B. 2 C. 16 D.Compiler error

[4]. Suppose List<String> list = new ArrayList<>(); Which of the following operations are correct?

```
A. list.add(new Integer(100));
B. list.add(new ArrayList());
C. list.add("Red");
D. list.add(new java.util.Date());
```

[5]. To find a maximum object in an array of strings (e.g., String[] names = {"red", "green", "blue"}), use

```
A. Arrays.max(names);
B. Collections.max(Arrays.asList(names));
C. Arrays.sort(names);
D. Collections.max(names);
```

[6]. This characteristic of object-oriented programming allows the correct version of an overridden method to be called when an instance of a subclass is used to call it.

A. Inheritance B.Aggregation C. Association D. Polymorphism

[7].The default equals method implementation of class Object determines:

```
A. whether two references refer to the same object in memory.
B. whether two references have the same type.
C. whether two objects have the same instance variables.
D. whether two objects have the same instance variable values.
```

[8].Which of the following keywords allows a subclass to access a superclass method even when the subclass has overridden the superclass method?

A. base.
B. this.
C. public.
D. super.

[9].Using the protected keyword includes giving a member:

A. public access. B. package access. C. private access. D.block scope.

[10]. What is the output ?

```
class A {
      A(){
        System.out.print(" Welcome");
          }
       A(String str){
          System.out.print(str);
       }
      }

  public class subClass  {

   public static void main(String[] args) {
        new C(" Hello");
        new B();
     }
   }
```

```
class B extends A {
        B(){}
        B(String str){
           super(str);
        }
      }

   class C extends B {
       C(){
          super(" Hi");
       }
       C(String str){
          super(str);
       }
     }
```

A. Hello Hi B. Hello Welcome C. Hello Hello D.Hello Welcome Hi

Fill Answers in the following sheet (*USING CAPTIAL LETTERS ONLY*):

الرجاء تعبئة جدول الاجابة بحروف اللغة الانجليزية الكبيرة فقط. لن يتم التصحيح لمن يخالف التعليمات

Questions	1	2	3	4	5	6	7	8	9	10
Answers	C	B	B	C	B	D	A	D	B	B

Question#2[16 Marks]: **Read each statement below carefully. Place a T on the line if you think a statement it TRUE. Place an F on the line if you think the statement is FALSE.**

T	1	Creating an array of objects references (an array of String object for example) does not create the actual objects whose references will be stored in the array.
T	2	Arrays themselves are objects, so an array variable is a reference to the actual array, therefor, two different array variables can be used refer to the same array memory.
F	3	Each object that belongs to a class has its own copy of static variables, and the values of these copies can change independently(مستقل) of the values of other copies.
T	4	A constructor is a special method with the same name as the class that is used to initialize the members of a class object. Constructors are called when objects of their classes are instantiated.
F	5	An instance method cannot refer to a static member (method or variable).
T	6	A method declared static cannot access non-static members and cannot invoke non-static method directly. A static method cannot use the this reference because static class variables and static methods exist independent of any object of a class.
T	7	Overloaded methods can have different return values, and must have different parameters lists. Two methods differing only by return type will result in a compilation error.
F	8	Constructors must declare a return type (either void or some other type).

Question#3 [20 Marks]

Write only one method that accepts the length of password as a parameter and generate a random password which includes characters from the following list(0-9,A-H,w-z, and $@!#%&), The signature of method is:

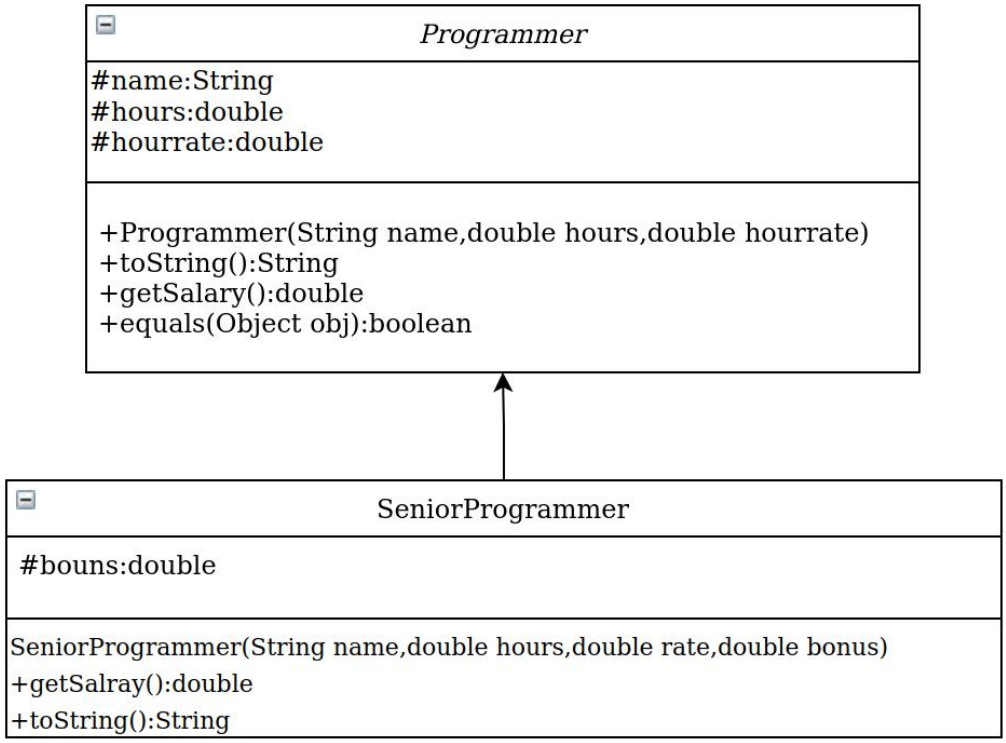
```
//Using Random Class


   public static String passwordRandomGenerator (int len) {
         String chars = "0123456789ABCDEFGHwxyz!@#$%&";
         Random rnd = new Random();
         StringBuilder sb = new StringBuilder(len);
         for (int i = 0; i < len; i++)
         sb.append(chars.charAt(rnd.nextInt(chars.length())));
         return sb.toString();
      }

 //Or using Math.random() method

   public static String passwordRandomGenerator (int len) {
      String chars = "0123456789ABCDEFGHwxyz!@#$%&";
      StringBuilder sb = new StringBuilder(len);
      for (int i = 0; i < len; i++)
   sb.append(chars.charAt((int)(Math.random()*chars.length())));
      return sb.toString();
   }
```

Question#4[40 Marks]: Given the following UML diagram



NOTE:

Assume that ALL setters and getters are given, no need to implement any of them
There is no grade for implementation of them.

[15%]
A. Write a Programmer class. The Programmers work number of hours per month. They should have an hourly rate (أجر) and number of hours per month, if they worked more than 140 hours per month then their total monthly salary will be raised up by 5%. **You have to Override the toString()** method to print programmer name with his/her salary. **Override the equals method** to return true if two programmers have the same salary.

```
class Programmer{
   protected String name;
   protected double hours;
   protected double rate;

   public Programmer(String name,double hours,double rate) {
         this.name = name;
         this.hours=hours;
         this.rate=rate;   }

   public String getName() {return name;}

   public void setName(String name) {this.name = name;}

   public double getHours() {return hours;}

   public void setHours(double hours) {  this.hours = hours;  }

   public double getRate() {return rate;}

   public void setRate(double rate) { this.rate = rate; }

   public double getSalary() {
      if(getHours()>140)
         return getRate()*getHours()*1.05;
    return getRate()*getHours();
    }
   @Override
   public String toString() {
      return "Programmer name=" + name + ", " + getSalary()  ;   }

   @Override
   public boolean equals(Object obj) {
      if(obj instanceof SeniorProgrammer)

        return (this.getSalary()==((SeniorProgrammer) obj).getSalary());
      else if (obj instanceof Programmer)
         return  (this.getSalary()==((Programmer) obj).getSalary());

      return false;  }


}
```

[15%]

B. Write a `SeniorProgrammer class` (مبرمج خبير). They have a bonus amount which is added as part of their salary. **Override the toString()** method to print programmer names with their salary and bonus.

```
class SeniorProgrammer extends Programmer{
   protected double bonus;

public SeniorProgrammer(String name,double hours,double rate,double bonus)
      {
       super(name,hours,rate);
      this.bonus = bonus;
   }

  public double getBonus() {
         return bonus;  }

   public void setBonus(double bonus) {
      this.bonus = bonus;  }


   public double getSalary() {
      return getBonus()+super.getSalary();
   }


   @Override
public String toString() {
      return "Senior Programmer name " +super.getName()+", Salary: "
+getSalary() ;
   }


}
```

[10%]
C. Write a driver class called `ProgrammerMain` that creates an `ArrayList` of programmers. Add 2 `SeniorProgrammers` and 2 Programmers. Write a method called **totalSalaries()** that takes an `ArrayList` of programmers as a parameter and calculates the sum of all programmers salaries. Then call this method from main and print the result.

```
public class PogrammerDriver {

   public static void main(String[] args) {

    ArrayList<Programmer> list =new ArrayList<>();

    list.add(new Programmer("Ahmad",140,10));
    list.add(new SeniorProgrammer("Ali",155,10,500));
    list.add(new Programmer("Zaid",140,10));
    list.add(new SeniorProgrammer("Sami",150,8,700));

    double total=totalSalary(list);
    System.out.println("Total Salaries= "+total);



   }


   public static double  totalSalary(ArrayList<Programmer> list) {
       double total=0.0;
      for(int i=0;i<list.size();i++) {
      total+=(list.get(i)).getSalary();

          }

     return total;
   }


    }
```