

COE 211 – Computer Programming

Welcome to Exam I
Tuesday March 13, 2018

Instructors: Dr. Dima El-khalil
Dr. Wissam F. Fawaz

Name: _____

Student ID: _____

Instructions:

1. This exam is **Closed Book**. Please do not forget to write your name and ID on the first page.
2. You have exactly **105 minutes** to complete the **5** required problems.
3. Read each problem carefully. If something appears ambiguous, please write your assumptions.
4. Do not get bogged-down on any one problem, you will have to work fast to complete this exam.
5. Put your answers in the space provided only. No other spaces will be graded or even looked at.

Good Luck!!

Problem 1: Multiple choice questions (20 minutes) [14 points]

For the questions given below, consider the following driver and helper classes:

```
import java.text.DecimalFormat;
public class Geometry {
public static void main(String[] args) {
Circle c1 = new Circle();
Circle c2 = new Circle();
System.out.println(c1);
c2.setArea(4.578);
System.out.println(c2.getPerim());
DecimalFormat fmt = new
DecimalFormat("0.##");
String area = fmt.format(c2.getArea());
area += 9;
System.out.println(
fmt.format(Double.parseDouble(area)));
System.out.println(fmt.format(c1.getArea()
+c2.getArea()));}
}
```

```
public class Circle {
private double radius, area, perim;
public Circle() {
radius = 2; area=0; perim=0;}
public double getRadius(){return radius; }
public void setArea(double a){area = a; }
public double getArea() { return area; }
public double getPerim() { return perim; }
public void calculatePerim() {
double perimeter = 3.14*2*radius;
perim = perimeter;
}
public String toString() {
return Double.toString(getRadius());
}
}
```

- 1) What output does the first `println` statement of the driver class produce?
 - a. 2
 - b. 0
 - c. 3.14
 - d. **None of the above**
- 2) What output does the second `println` statement of the driver class produce?
 - a. 2
 - b. 0
 - c. 3.14
 - d. **None of the above**
- 3) What output does the third `println` statement of the driver class produce?
 - a. 4.60
 - b. 4.58
 - c. **4.59**
 - d. None of the above
- 4) What output does the fourth `println` statement of the driver class produce?
 - a. 4.60
 - b. **4.58**
 - c. 4.59
 - d. None of the above
- 5) How many local variables are created inside the main method of the driver class?
 - a. **4**
 - b. 3
 - c. 2
 - d. None of the above
- 6) How many constructors does the driver class have?
 - a. 0
 - b. **1**
 - c. 2
 - d. None of the above
- 7) How many local variables are used inside the helper class?
 - a. 0

- b. 1
 c. **2**
 d. None of the above
- 8) Consider the body of the `getPerim` method defined inside the helper class. Which of the following can be used to store a more accurate value of the perimeter in the `perimeter` variable? Note that the perimeter of a circle is $2\pi r$
- a. `2 Math.PI * R`
 b. `2 Math.PI() * R`
 c. **`(11/5) * Math.PI * R`**
 d. None of the above
- 9) Consider the `c1` object created in the driver class. Which of the following statements correctly finds its area and stores it in `c1`'s `area` instance variable? Note that the area of a circle is πr^2
- a. `c1.setArea(c1.calculatePerim()*c1.getRadius()/2);`
 b. **`c1.setArea(Math.PI*Math.pow(c1.getRadius(), 2));`**
 c. Both of the above
 d. None of the above
- 10) Consider the `c2` object created in the driver class. Which of the following statements correctly finds the square root of the area of `c2` and then stores it in a `String` variable called `str`?
- a. `String str=Math.sqrt(c1.getArea())+"";`
 b. `String str=Integer.toString(Math.sqrt(c2.getArea()));`
 c. Both of the above
 d. **None of the above**
- 11) Assuming that the `str` variable from the previous question has been created correctly, which of the following can be used to extract the decimal point (.) that it may contain?
- a. `char point=str.charAt(str.indexOf('.')-1);`
 b. **`String point=str.substring(str.indexOf('.'),str.indexOf('.')+1);`**
 c. Both of the above
 d. None of the above
- 12) Which of the following statements correctly creates a variable called `diameter` that stores the diameter of `c1`?
- a. **`double diameter=2*Double.parseDouble(c1.toString());`**
 b. `double diameter=2*c1.radius;`
 c. Both of the above
 d. None of the above
- 13) Consider the `diameter` variable from the previous question, which of the following statements correctly prints out the lowest integer whole number that is greater than `diameter`? For example, if `diameter=23.5`, then the output should be 24.
- a. `System.out.print((int) diameter);`
 b. **`System.out.print((int) (diameter+1));`**
 c. Both of the above
 d. None of the above
- 14) Which of the following correctly prints out the perimeter of `c1`?
- a. **`c1.calculatePerim();`**
`System.out.print(c1.getPerim());`
 b. `System.out.print(c1.calculatePerim());`
 c. Both of the above
 d. None of the above

Problem 2: True or false questions (10 minutes) [11 points]

1. The following Java statement results in a compile time error:

```
System.out.println("\\\\");
```

 Answer: **True** **False**

2. Not providing a constructor for a class results in a compile-time error.
 Answer: **True** **False**

3. The following code fragment results in a run-time. Assume that the `DecimalFormat` class was imported properly.

```
DecimalFormat fmt = new DecimalFormat("0.#");
double value = 0.46; String str = fmt.format(value);
System.out.println(str.charAt(3));
```

 Answer: **True** **False**

4. The following statement prints out to the screen: 4

```
System.out.print(Math.pow(Math.sqrt(4), 2));
```

 Answer: **True** **False**

5. The following statement prints out a random `int` value between 1 (inclusive) and 5 (inclusive). Assume that `rnd` is a `Random` object that was instantiated properly.

```
System.out.print(Math.abs(rnd.nextInt())%5+1);
```

 Answer: **True** **False**

6. The following statement modifies the value of the `String` variable called `str` by replacing every occurrence of the letter `'a'` with the letter `'e'`.

```
str.replace('a', 'e');
```

 Answer: **True** **False**

7. The following code fragment correctly prints out the last 2 characters of the `String` variable called `str`

```
System.out.print(
str.charAt(str.length()-2)+str.charAt(str.length()-1));
```

 Answer: **True** **False**

8. The following assignment statement that stores an `int` literal value in a `Double` object called `obj` is syntactically valid; that is, it does not produce a compile-time error.

```
Double obj = 23;
```

 Answer: **True** **False**

9. All the methods that do not return any value must use `void` as their return type.
 Answer: **True** **False**

10. The `private` methods of a class are not allowed to access its instance variables.
 Answer: **True** **False**

11. The following Java statements increment the `int` variable called `val` by 3.

```
val += 2; val+ +;
```

 Answer: **True** **False**

Problem 3: Code analysis (15 minutes) [10 points]

1) Consider the helper class given below, along with a driver class for it.

<pre>public class ClassA { private String str; private int index; public ClassA(String s) { str=s; index=-1;} public void first(){ index=str.indexOf(':');} public int second() { String str1, str2; str1=str.substring(0,index-1); str2=str.substring(index+2); return third(str1,str2); } public int third(String s1,String s2){ return s1.length()+s2.length(); } }</pre>	<pre>public class ClassADriver { public static void main(String[] args) { String name1="Michael:Jordan"; String name2="Shaquille:Oneal"; ClassA obj1 = new ClassA(name1); ClassA obj2 = new ClassA(name2); obj1.first(); obj2.first(); int val1 = obj1.second(); int val2 = obj2.second(); System.out.print("Answer: " + val1+val2); } }</pre>
--	--

When running the ClassADriver class, what output is produced?

- Answer: 1112**
- Answer: 25
- An error occurs during the execution of the code
- It doesn't compile correctly
- None of the above

2) Consider the class given below, along with a driver class for it.

<pre>public class ClassB { private Integer obj1, obj2; public ClassB(int val1, int val2) { obj1 = val1; obj2 = val2;} public double first() { return obj1.intValue()%obj2.intValue(); } public double second() { return obj1.doubleValue()/obj2.intValue();} public String toString() { return Double.toString(first() + second());} }</pre>	<pre>public class ClassBDriver { public static void main(String[] args) { Double obj1 = 7.4, obj2 = 2.8; ClassB obj = new ClassB(obj1.intValue(), obj2.intValue()); String str = obj + ""; System.out.println("Answer: " + str); } }</pre>
--	---

When running ClassBDriver class, what output is produced?

- Answer: 5.7
- Answer: 4.5**
- An error occurs during the execution of the code
- It doesn't compile correctly
- None of the above

Problem 4: Evaluating Java expressions (20 minutes) [25 points]

For each of the following code fragments, what is the value of x after the statements are executed?

- (1)

```
String str = "aaab";
int index1 = (int) Math.random()*2+1;
int index2 = (int) (Math.random()*3);
char letter1=str.charAt(index1);
char letter2=str.charAt(index2);
String x = "" + letter1 + letter2;
```

Answer: x= "aa"
- (2)

```
double val1 = Math.floor(3.5);
double val2 = Math.ceil(4.7);
double x = Math.pow(val1, (int) val1/(int) val2);
```

Answer: x= 1.0
- (3)

```
String nbs = "589";
int z = Integer.parseInt(nbs.substring(0,2));
int y = Integer.parseInt(nbs.substring(2)) ;
int x = ++z ;
x -= (z+y) ;
```

Answer: x= -9
- (4)

```
DecimalFormat fmt = new DecimalFormat("00.##");
String str = fmt.format(2.597);
String x = str.substring(0, str.length()/2);
```

Answer: x= "02"
- (5)

```
String str1 = "Java is fun";
String str2 = str1.toUpperCase().toLowerCase();
boolean x = str1.equals(str2);
```

Answer: x= false
- (6)

```
double angle = (int) Math.PI/4;
double x = Math.cos(angle);
```

Answer: x= 1.0
- (7)

```
String str = "Dont worry, be happy";
char x = str.replace('o', 'a').charAt((str.length()-1)/3);
```

Answer: x= 'a'

- (8)

```
String str1 = "Just do it";
String str2 = str1.replace('u', 'o').replace('o', 'u');
boolean x = str2.equalsIgnoreCase(str1);
```

Answer: x= false
- (9)

```
String str = "Live free or die";
int index1 = str.indexOf('e');
int index2 = str.indexOf('r');
index1--;
index2++;
String x = str.substring(index1, index2);
```

Answer: x= "ve fr"
- (10)

```
double value = Math.pow(4, 1/2);
double x = Math.sqrt(value*value);
```

Answer: x= 1.0
- (11)

```
boolean flag = false;
String str = flag + "";
char x = str.toUpperCase().charAt(str.length()/2);
```

Answer: x= 'L'
- (12)

```
String S1 = new String("****///");
String S2 = S1.length() + 2 + "2" ;
boolean x = S2.equals("102");
```

Answer: x= true

Problem 5: Coding Problems (40 minutes) [40 points]

1. Write a Java program called `InsertSpace` that reads from the user a String `S` and an `int` value representing a valid index within `S`. Your program should then create a modified version of `S` called `S1` by inserting into `S` a tab character at the user-supplied index value. After creating `S1`, your program is required to print it out to the screen.

Sample run:

Please enter a String: COE211-Computer Programming

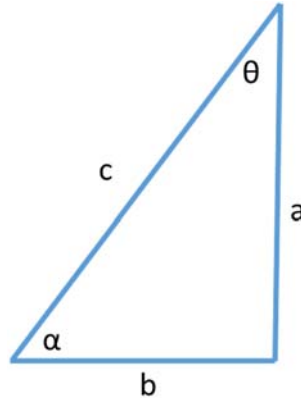
Please enter the index of tab: 3

The modified String: COE 211-Computer Programming

```
import java.util.Scanner;
public class InsertSpace {
    public static void main (String[] args) {
        Scanner scan= new Scanner(System.in);
        System.out.print("Please enter a String: ");
        String S=scan.nextLine();
        System.out.print("Please enter the index of tab: ");
        int a=scan.nextInt();
        String S1=S.substring(0,a)+"\t"+S.substring(a);
        System.out.print("The new String: "+S1);
    }
}
```


2. Consider the right triangle given in the figure below. Write a Java program that reads from the user the length of its hypotenuse c . Your program should then generate a random degree value for one of its acute angles, namely θ (see the figure below). Your program should then use the randomly generated value θ to derive and print out to the screen the lengths of the sides a and b rounded to 4 decimal places. Note that a and b are related to c as follows:

$$b = c \sin(\theta) \quad \text{and} \quad a = c \cos(\theta).$$



Sample run:

Please enter the length of the hypotenuse: 10

Acute angle value: 20

a = 9.3969

b = 3.4202

```
import java.util.Random;
import java.util.Scanner;
import java.text.DecimalFormat;

public class RightTriangle {

    public static void main(String[] args) {
        Scanner scan= new Scanner(System.in);
        System.out.print("Please enter the length of the hypotenuse: ");

        double H=scan.nextDouble();

        Random rnd = new Random();
        double angle=(rnd.nextFloat()*90);

        double S1= H*Math.cos(angle*Math.PI/180);
        double S2= H*Math.sin(angle*Math.PI/180);

        DecimalFormat fmt= new DecimalFormat("0.####");

        System.out.println("Acute angle value: " + angle);
        System.out.println("a = "+fmt.format(S1));
        System.out.println("b = "+fmt.format(S2));

    }
}
```

- For a particular car, repair and maintenance costs in year 1 are estimated at x \$ and in year 10 at y \$, with x and y being two `int` values. Assuming that the repair cost increases by the same amount every year, write a Java application that asks the user to enter the value of x , the value of y , and a year number. Then, your program should compute and display on the screen the repair and maintenance costs for that year rounded to two decimal places.

Sample run:

Enter repair and maintenance cost for year 1 in \$: 100

Enter repair and maintenance cost for year 10 in \$: 1500

Which year would you like to know the cost for? 5

The cost for year 5 is \$722.22.

```
import java.text.NumberFormat;
import java.util.Scanner;
public class TestClass {
public static void main(String[] args){
NumberFormat fmt = NumberFormat.getCurrencyInstance();
int x, y, yearNb;
double costIncrement, totalCost;
Scanner scan = new Scanner(System.in);
System.out.print("Enter repair and maintenance cost for year 1 in $:");
x = scan.nextInt();
System.out.print("Enter repair and maintenance cost for year 10 in $:");
y = scan.nextInt();
System.out.print("Which year would you like to know the cost for? ");
yearNb = scan.nextInt();
costIncrement = (y-x) / 9.0;
totalCost = x + costIncrement*(yearNb-1);
System.out.println("The cost for year 5 is " + fmt.format(totalCost));
}}
```

4. Write a program called `RandomWordGenerator` that creates a `String` object `S` consisting of the characters "abercd" in this order. Your program should then create and print out a new `String` `S1` consisting of exactly 4 unique characters that are randomly selected from `S`. Note that `S1` is not allowed to have duplicate characters and should be composed of distinct characters only. For instance, the value "bbae" is not a valid value for `S1` since it contains two "b" characters. On the other hand, "baer" is a valid value for `S1` as no two characters are identical in that value.

Sample run 1:

Output string S1: bear

Sample run 2:

Output string S1: care

```
import java.util.Random;
public class RndWordGenerator {
public static void main(String[] args) {
    String str = "abercd";
    Random rnd = new Random();
    String str1 = "";
    int rndIndex;
    rndIndex = rnd.nextInt(str.length());
    str1 = str1 + str.charAt(rndIndex);
    str = str.substring(0, rndIndex) + str.substring(rndIndex+1);
    rndIndex = rnd.nextInt(str.length());
    str1 = str1+str.charAt(rndIndex);
    str = str.substring(0, rndIndex) + str.substring(rndIndex+1);
    rndIndex = rnd.nextInt(str.length());
    str1 = str1 + str.charAt(rndIndex);
    str = str.substring(0, rndIndex) + str.substring(rndIndex+1);
    rndIndex = rnd.nextInt(str.length());
    str1 = str1 + str.charAt(rndIndex);
    str = str.substring(0, rndIndex) + str.substring(rndIndex+1);
    System.out.println("Output string S1: " + str1);
}
}
```