United International University (UIU)

Dept. of Computer Science and Engineering (CSE)

FINAL EXAM :: FALL 2018

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

Date: **09/01/19** Total Marks: **40** Time: **2 Hours**

1. a) Declare a static nested class **Inner1** of an outer class **Outer1** and a nested class **Inner2** of an outer class **Outer2.** Now, create object of **Inner1** and **Inner2** class respectively inside **NestedDemo** class's main method. [2+2]

   b) Rewrite this program using anonymous inner class. [4]

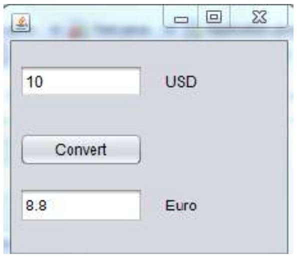
```
public interface Fruit {
    int quantity=10;
    void isRipe();
}
public class Apple implements Fruit{
    public void isRipe() {
        System.out.println("Apples are ripe");
    }
}
```

```
public class Test {
    public static void main(String[]
args)
    {
        Apple a=new Apple();
        a.isRipe();
    }
}
```

2. The records of Chikungunya victims are stored in a txt file name **victims.txt**. Each record contains name followed by age followed by district; the record are stored in ascending order of district name. Write a program that will read the records from the file, find the district with highest Chikungunya victims, and display the district name with victim count in console. [8]

victims.txt	**Expected Output**
Bari-45-Barishal Hasan-50-Barishal Kayes-25-Dhaka Rawnak-35-Dhaka Jabed-43-Dhaka Sumi-48-Rajshahi	Dhaka-3

3. You are required to complete a Java GUI application like below that has the functionality of converting **US Dollar** to **Euro** after pressing the **Convert** Button. Formula: 1 USD = 0.88 Euro. [8]



10 USD

8.8 Euro

4. a) Suppose you have a class named **Employee**. The **Employee** class has three members: employeeName, employeeId and salary. Now: [4]

   i) Create an ArrayList of type **Employee**
   ii) Create three objects of **Employee** class and add those to the ArrayList
   iii) Sort the ArrayList according to **Employee** salary
   iv) Print the sorted ArrayList

b) Write the output of the following program: [4]

```
public class MyThread extends Thread{
    MyThread() {
        System.out.print("MyThread");
    }
    public void run() {
      for(int i = 0; i< 4; i++) {
            System.out.println(" running");
            try {sleep(1000);}
            catch (InterruptedException e) {}
      }
    }
    public void run(String s) {
      System.out.print(s + " is running again");
    }
}
```

```
public class Application {
    public static void main (String []
args) {
        Thread t = new MyThread() {
            public void run() {
            System.out.println(" are you
running?");
            }
        };

        ((MyThread)t).run("MyThread");
        t.start();

    }
}
```

5. a) Fix the errors in the following code. You cannot add or remove any functions. [2]

```
public class Test {
    static String str = "+";
    public static void main(String[] args) {f1(); }
    static void f1(){
        try{
          f2();
          throws Exception();
        }
        catch (Exception e){ str += "-";}
    }
    static void throws Exception f2(){
        throw new Exception();
    }
}
```

b) Write the output of the given code. [3]

```
public class Main {
    static String str = "a";
    public static void main(String[] args){
      try{
        str += "b";
        System.out.println(str);
        throw new Exception("Whatever");
      }
      catch (Exception e){str += "c"; }
      finally{
        str += "d";
        System.out.println(str);
        str += "e";
      }
      System.out.println(str);
    }
}
```

c) Identify the type of Exception in the given code. Use appropriate catch block(s) to handle it. [3]

```
public class MyException {
    public static void main(String[] args){
        int a[]=new int[5];
        a[5]=10;
        Integer.parseInt("abc");
        Scanner scan=new Scanner(System.in);
        int x=scan.nextInt();
    }
}
```