

Family Name: Other Names:

Student ID: Signature

COMP 102/112: Test

2021, 14 June ** WITH SOLUTIONS **

Instructions

- Time allowed: **120 minutes**
- Attempt **all** the questions. There are 120 marks in total.
- Write your answers in this test paper and hand in all sheets. (We have different instructions for distance students)
- If you think a question is unclear, ask for clarification.
- Brief Java documentation is provided with the test.
- This test contributes 50% of your final grade.
- You may use dictionaries and calculators.
- You may write notes and working on this paper, but make sure your answers are clear.
- You may assume all the programs import the ecs100 library and other standard libraries.

Questions:

1. Java basics [25]
2. Design a class [15]
3. Files [23]
4. ArrayLists of objects [34]
5. 2D arrays [23]

June 4, 2021

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

Question 1. Java basics**[25 marks]****(a) [8 marks] Conditionals**

Write a method `shippingFee(...)` that calculates the shipping cost for a delivery address. The method should have a parameter to specify the address, and returns the cost as follows:

- Local: \$10.00
- Nation-wide: \$20.00
- International: \$50.00

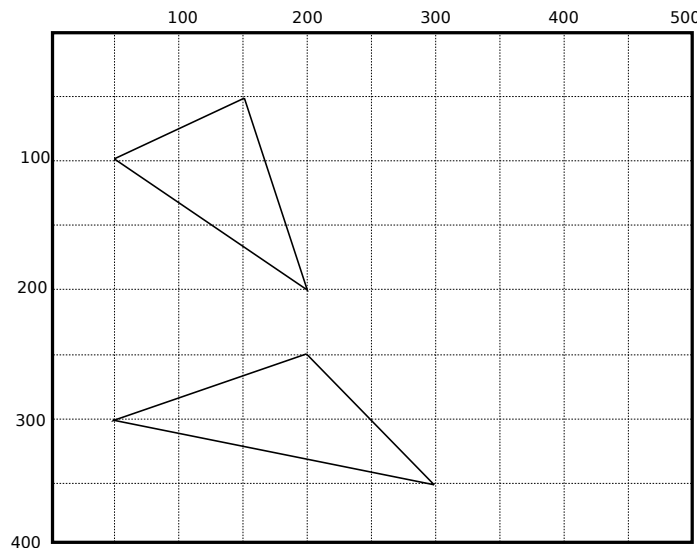
The address is a string and it always contains "Wellington" for local, and contains "New Zealand" or "NZ" for nation-wide. You may assume any other address strings are international.

```
public double shippingFee(String address) {  
  
    address.toLowerCase();  
    if (address.contains("wellington"))  
        return 10;  
    else if (address.contains("new zealand") || address.contains("nz"))  
        return 20;  
    else  
        return 50;  
}  
  
}
```

(b) [9 marks] Define a method and call it in the same class.

Complete the drawTriangle(...) method below to draw a triangle in the graphics pane. It should have parameters that specify the coordinates of the three points of the triangle.

Complete the draw2Triangles() method to call the drawTriangle(...) method twice to draw the picture shown below:



```

public void drawTriangle(
    double x1, double y1, double x2, double y2, double x3, double y3
) {

    UI.drawLine(x1, y1, x2, y2);
    UI.drawLine(x2, y2, x3, y3);
    UI.drawLine(x3, y3, x1, y1);

}

public void draw2Triangles( ) {

    this.drawTriangle(150, 50, 200, 200, 50, 100);
    this.drawTriangle(200, 250, 50, 300, 300, 350);

}

```


Question 2. Design a class**[15 marks]**

Suppose you are writing a program to allow users to list their used items for sale. Item should have a name, a price and a list of features, e.g.

- name: "snowboard"
- price: 200.0
- features: "good condition", "150cm", "pickup only"

Complete the Item class on the facing page to specify Item objects.

- The fields should store the name, price and a list of features of the item.
- The constructor should create an Item object with the specified name and price. It should also create a new list in order to hold the features.
- The addFeature(...) method should allow the user to add a specific feature to the item.
- The print method should print all the information in the fields, with each feature on a separate line.

```
public class Item {  
    // Fields  
  
    private String name ;  
    private double price ;  
    private ArrayList<String> features;  
  
    //Constructor  
    public Item(String name, double price) {  
  
        this.name = name;  
        this.price = price;  
        this.features = new ArrayList<String>();  
  
    }  
    public void addFeature(String s){  
  
        this.features.add(s);  
  
    }  
  
    public void print(){  
        UI.println (this.name);  
        UI.println (this.price);  
        for (String s: this.features) {  
            UI.println (s);  
        }  
  
    }  
}
```

Question 3. Files**[23 marks]**

A data file car.txt is shown below. Each line contains a car make, model and two integers for volume and weight.

```
Toyota Aygo 1000 790
Mitsubishi Space Star 1200 1160
Fiat 500 900 865
BMW 1 1600 1365
Mercedes A-Class 1500 1365
Mazda 323 2200 1280
BMW 5 2000 1705
Ford Crown Victoria 4200 1873
```

(a) **[8 marks]** What will the following method print out?

```
public void testingFile () {

    try {
        List<String> lines = Files.readAllLines (Path.of("car.txt"));

        for (String line : lines) {
            Scanner scan = new Scanner(line);
            scan.next();
            scan.next();
            UI.println (scan.next());
        }

    } catch (IOException e) {UI.println ("File error"); }
}
```

```
1000
Star
900
1600
1500
2200
2000
Victoria
```


(b) [15 marks] Complete the following printModel() method that will read the file, extract the model in each line and print them on separate lines.

Please note that the car make is always one word (the first word on each line), but the model can be one word, two words or a single integer number. The rest of the line (volume and weight) are always integers.

For example, if the file car.txt contains data showing on the facing page, your method should print out

```
Aygo
Space Star
500
1
A-Class
323
5
Crown Victoria
```

```
public void printModel() {
    try {

        List<String> lines = Files.readAllLines(Path.of("car.txt"));

        for (String line : lines) {
            Scanner scan = new Scanner(line);
            scan.next();
            String model="";
            if (scan.hasNextInt()){
                model = scan.next();
                UI.println (model);
            }else {
                model = scan.next();
                if (!scan.hasNextInt()){
                    model = model + " " + scan.next();
                }
                UI.println (model);
            }
        }

    } catch (IOException e) {UI.println ("File error");}
}
```

Question 4. ArrayLists of Objects**[34 marks]**

The program has two classes: a ShoeStore class and a Shoe class.

The Shoe class below defines Shoe objects.

```
public class Shoe {
    private String brand;
    private int size;           //The shoe size is European size.
    private double price;

    public Shoe(String b, int s, double p) {
        this.brand = b;
        this.size = s;
        this.price = p;
    }

    public String getBrand() {
        return this.brand;
    }

    public int getSize() {
        return this.size;
    }

    public double getPrice() {
        return this.price;
    }
}
```

(Question 4 continued)

The ShoeStore class declares a shoes field to store the list of Shoe objects:

```
private ArrayList<Shoe> shoes = new ArrayList<Shoe>();
```

(a) [8 marks] What will the following method print out?

```
public void testing () {  
    Shoe a = new Shoe("converse", 42, 120.00);  
    Shoe b = new Shoe("ecco", 40, 260.00);  
    Shoe c = new Shoe("vans", 38, 100.00);  
    Shoe d = new Shoe("nike", 42, 250.00);  
    this.shoes.clear ();  
    this.shoes.add(a);  
    this.shoes.add(new Shoe("adidas", 42, 200.00));  
    this.shoes.add(0, b);  
    this.shoes.add(c);  
    this.shoes.add(1,d);  
    for(Shoe m: shoes){  
        Ul.println (m.getBrand());  
    }  
  
    Ul.println (this.shoes.get(0).getSize ());  
}
```

```
ecco  
nike  
converse  
adidas  
vans  
40
```

Student ID:

(Question 4 continued on next page)

(Question 4 continued)

(b) [14 marks] The following findShoe(...) method has two parameters which are the brand and the size.

Complete the findShoe(...) method that will search the list of shoes to find all shoes in the particular brand and size, and print their prices.

If nothing is found, then print a "not found" message.

```
public void findShoe(String brand, int size) {  
  
    boolean exists = false;  
    for(Shoe m : shoes) {  
        if(m.getBrand().equals(brand) && m.getSize()==size) {  
            UI.println (m.getPrice());  
            exists = true;  
        }  
    }  
    if (!exists) {  
        UI.println ("Not found");  
    }  
  
}
```

Student ID:

(Question 4 continued on next page)

(Question 4 continued)

(c) [12 marks] Complete the following `returnOneSize(...)` method that will find all the shoes in one particular size, save them in a new `ArrayList`, and return the `ArrayList`.

```
public ArrayList<Shoe> returnOneSize(int size) {  
  
    ArrayList<Shoe> oneSize = new ArrayList<Shoe>();  
  
    for (Shoe s : shoes) {  
        if (s.getSize()==size) {  
            oneSize.add(s);  
        }  
    }  
    return oneSize;  
  
}
```

SPARE PAGE FOR EXTRA ANSWERS

Cross out rough working that you do not want marked.
Specify the question number for work that you do want marked.

Question 5. 2D Arrays**[23 marks]**

The matrix field is declared to hold a 2D array as follows:

```
private int [][] matrix = new int[][]{{ 10, 20, 31, 11, 11, 11, 25},
                                       { 15, 15, 15, 15, 14, 14, 14},
                                       { 14, 14, 14, 14, 10, 20, 17},
                                       { 30, 20, 22, 22, 22, 22, 22},
                                       { 20, 11, 11, 11, 16, 19, 30}};
```

(a) **[8 marks]** What will the following method print out?

```
public void testing2DArray() {
    Ul.println (this.matrix [0][3]);
    Ul.println (this.matrix.length);
    Ul.println (this.matrix [2]);
    for(int j=0; j < 3; j++) {
        Ul.println (this.matrix[j ][2]);
    }
}
```

```
11
5
[14, 14, 14, 14, 10, 20, 17]
31
15
14
```

(Question 5 continued)

(b) [15 marks] Complete the following `findLongestRepeat()` method that will find the longest sequence of duplicate numbers in each row of the array in the matrix field.

For example, if the 2D array is declared as follows: (the same as part a)

```
private int [][] matrix = new int[][]{{ 10, 20, 31, 11, 11, 11, 25},
                                       { 15, 15, 15, 15, 14, 14, 14},
                                       { 14, 14, 14, 14, 10, 20, 17},
                                       { 30, 20, 22, 22, 22, 22, 22},
                                       { 20, 11, 11, 11, 16, 19, 30}};
```

`findLongestRepeat()` should print out the following:

```
5 * 22 is at row 3, from 2 to 6
```

Please note that:

- 5 is the length of the longest sequence
- 22 is the repeated value
- 3 is the row index
- 2 is the start column index
- 6 is the end column index

Your method should still work whatever the size of the array is, as long as it contains at least one value.

```

public void findLongestRepeat(){
    int countMax = 0;
    int startMax = 0;
    int endMax = 0;
    int rowMax=0;
    int valueMax = this.matrix [0][0];

    for (int i = 0; i < this.matrix.length; i++) {

        int value = this.matrix[i ][0];
        int count = 1;
        int start = 0;
        int end = 0;

        for(int j=1; j < this.matrix[i ].length; j++) {
            if (this.matrix[i ][j] == value){
                count++;
                end = j;
                if (count > countMax){
                    countMax = count;
                    valueMax = value;
                    rowMax = i;
                    startMax = start;
                    endMax = end;
                }
            }
            else{
                count = 1;
                value = this.matrix[i ][j];
                start = j;
            }
        }
    }
    Ul.println (valueMax + " * " + countMax + " at: " + rowMax + " from "+ startMax + " to "+ e

}

}

```
