Family Name:. . . . .   Other Names: . . . .

Student ID:. . . . .   Signature . . . .

COMP 102/112: Test

2019, May 14 ** WITH SOLUTIONS **

Instructions

- Time allowed: **50 minutes**
- Attempt **all** the questions. There are 50 marks in total.
- Write your answers in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- Brief Java documentation is provided with the test
- This test contributes 15% of your final grade
  (But your mark will be increased to your exam mark if that is higher.)
- You may use dictionaries.
- You may write notes and working on this paper, but make sure your answers are clear.

Questions	Marks	
1. Understanding while Loops	[4]	
2. Files	[8]	
2. Defining Classes	[16]	
3. Event Driven Input	[12]	
4. ArrayLists	[10]	
	TOTAL:	

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. Understanding while Loops   **[4 marks]**

What will the following printData method print out?.

```
public void printData(){
    int c = 0;
    while(c < 10){
        UI. println (c + ": " + c * 10);
        c = c + 2;
    }
}
```

```
0:   0
2:   20
4:   40
6:   60
8:   80
```

Mark:

Question 2. Files [8 marks]

Suppose a file is simply a sequence of numbers. Complete the printNumbersInRange method on the facing page that will find and print all numbers in a file that are in the range [n1, n2], where n1 and n2 are specified in the parameters.

For example, if the file data.txt contains:

```
2 5 6 8 3 5 1 9 7 4 3 2 2 9 3 5 7 5 3
```

printNumbersInRange(3, 5, "data.txt") should print out

```
5.0 3.0 5.0 4.0 3.0 3.0 5.0 5.0 3.0
```

The method readAllNumbers is done for you. You can call it to read all the numbers in a file into an ArrayList.

(Question 2 continued)

```java
public void printNumbersInRange(int n1, int n2, String filename){

    ArrayList<Double> nums = readAllNumbers(filename);
    for (double n : nums) {
        if (n >= n1 && n <= n2){
            UI.print (n + " ");
        }
    }


















}
public ArrayList<Double> readAllNumbers(String filename){
    ArrayList<Double> nums = new ArrayList<Double>();
    try{
        Scanner scan = new Scanner(new File(filename));
        while (scan.hasNextDouble()){
            nums.add(scan.nextDouble());
        }
        scan. close ();
    }
    catch(Exception e) {UI. println ("File error");}
    return nums;
}
```

Mark:

Question 3. Defining Classes [16 marks]

This question is about a program that will draw trees on the graphics pane, make the trees grow and be trimmed.

(a) **[10 marks]** Design a Tree class to represent Tree objects. The class contains:

- Data fields that store the size and the location of the tree.

  Suppose you are using a filled grey rectangle to draw the trunk and a filled green circle to draw the crown. All trees start with the same default size.

  You need to make sure that the root of the tree does not move when the tree grows.

- A constructor that initializes the fields. You are required to initialize the position of the tree. Other fields should use default values.

- The draw method that draws the tree at the specified location with the right size.

- The getCrownSize method that returns the diameter of the crown.

- The trim method that reduces the diameter of the crown to half of its original size.

- The grow method that increases the height of the trunk and the diameter of the crown by 10 pixels, and increases the width of the trunk by 2 pixels.
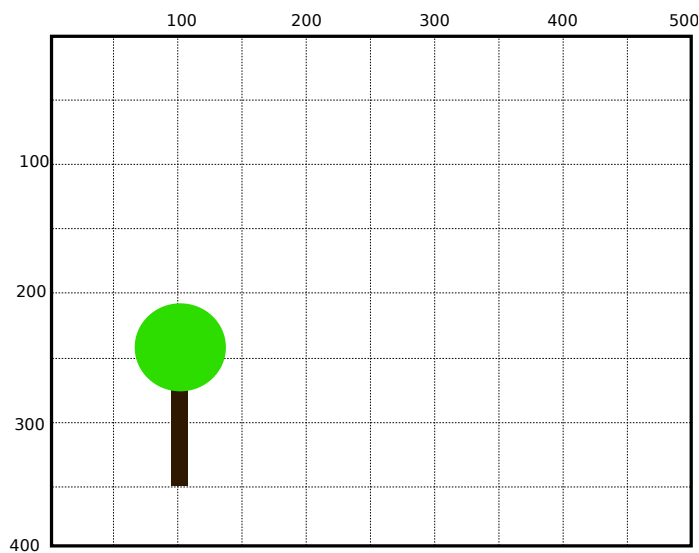
```java
class Tree {

    private double width = 6;
    private double height = 20;
    private double crownSize = 15;
    private double x;
    private double y;
    public Tree(double x, double y){

        this.x = x;
        this.y = y;


    }
    public void draw(){

        UI.setColor(Color.gray);
        UI.fillRect(x−width/2, y−height, this.width, height);
        UI.setColor(Color.green);
        UI.fillOval(x−this.crownSize/2, y−height−this.crownSize/2, this.crownSize, this.crownSize);



    }
    public double getCrownSize(){


        return this.crownSize;


    }
    public void grow(){
        this.height = this.height + 10;
        this.width = this.width + 2;
        this.crownSize = this.crownSize + 10;



    }
    public void trim(){

        this.crownSize = this.crownSize/2;

}
    }
}
```

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.

(b) **[6 marks]** Complete the following testTree method in the Garden class that will create a Tree object at (100, 400), draw it in the graphics pane, make it grow and be trimmed. Write a loop to make the tree grow repeatedly for at least 40 times. During the growing, it should be trimmed whenever its crown reaches a diameter of 100.

```java
public class Garden{
    public void testTree(){
        Tree t = new Tree(100, 400);


        for ( int i = 0; i < 20; i++){
            t.grow();

            if (t.getSize() >= 100){
                t.trim ();
            }
            UI.sleep (500);
            UI.clearGraphics ();
            t.draw();
        }


    }
}
```

Mark:

Question 4. Event-Driven Input   **[12 marks]**

Complete the CircleAndRing program on the facing page so that it allows the user to draw circles and rings with different sizes on the graphics pane.

The program should have one button which allows the user to swap between "circle" and "ring", and one slider which allows the user to set the diameter of the circle/ring to be drawn.

A ring should be drawn as two circles, where the outside circle is drawn using the specified diameter and the inside circle is drawn using a half sized diameter.

When the user releases the mouse at a position on the window, the program should draw a circle or ring of the specified size centered at that position.
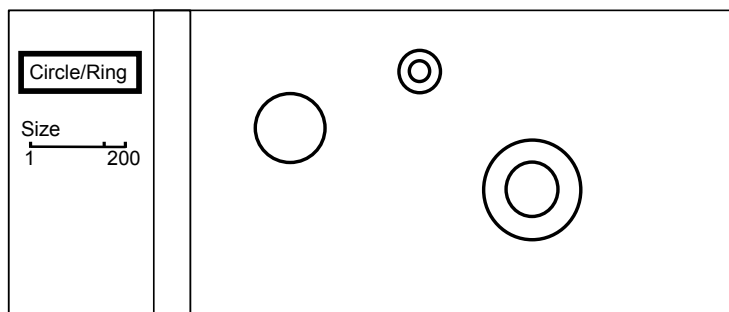
Hint: The addSlider method requires a name, a minimum value, a maximum value, an initial value and a method to call.

(Question 4 continued)

```java
public class CircleAndRing {
  // fields

   private double size = 20; // size of the circles
   private boolean circle = true;
  //Define methods for the button and the slider

   public void setShape(){
      this.circle = !this.circle;
      /* this is the same as
      if (this.circle)
          this.circle = false;
      else
          this.circle = true;
      */
   }



   public void setSize(double v){
      this.size = v;
   }
  //Complete the method for mouse
  public void doMouse(String action, double x, double y){

      if (action.equals("released")){
          UI.drawOval(x−size/2, y−size/2, size, size);
          if (!this.circle){
              UI.drawOval(x−size/4, y−size/4, size/2, size/2);
          }
      }



   }
  public static void main(String[] arguments){
      CircleAndRing cr = new CircleAndRing();
      UI.addButton("circle/ring", cr::setShape);
      UI.addSlider("size", 1, 100, 20, cr::setSize);
      UI.setMouseListener(cr::doMouse);



   }
}
```

Mark:

Question 5. ArrayLists [10 marks]

Suppose the data.txt file contains a sequence of numbers. Complete the findLongestEqualValueSequence method on the facing page that will find the longest sequence of **equal** values, print the value of the number in this sequence and the length of this sequence.

For example, if the file contains "2 2 3 5 3 3 3 9 1 2 3 5 5 5 5 5 5 5 4 4 4 2 2 1 1", your method should print:

```
The value is : 5.0
The length is : 7
```

The first line (done for you) calls the readAllNumbers method given in Question 2 to load the numbers into an ArrayList. You do not need to define the readAllNumbers method.

If there are two sequences having the same length, print information for the first sequence.

(Question 5 continued)

```java
public void findLongestEqualValueSequence(){
    ArrayList<Double> nums = readAllNumbers("data.txt");
    double pre_n = nums.get(0);
    int count = 1;
    int maxCount = 0;
    double maxCountNum = 0;

    for(double n: nums){
        if (n == pre_n){
            count++;
            if (count > maxCount){
                maxCount = count;
                maxCountNum = n;
            }
        }
        else{
            count = 1;
        }
        pre_n = n;
    }
    UI. println ("The value is: " + maxCountNum);
    UI. println ("The length is " + maxCount);



}
```

Mark:

* * * * * * * * * * * * * *