

Family Name: .....

Other Names: .....

ID Number: .....

Signature .....

## COMP 102: Test 2

14 May, 2013

### Instructions

- Time allowed: **50 minutes**
- There are 45 marks in total.
- Answer **all** the questions.
- Write your answers in the boxes in this test paper and hand in all sheets. You may ask for additional paper if you need it.
- If you think some question is unclear, ask for clarification.
- Brief Java documentation will be supplied with the test.
- This test will contribute 15% of your final grade,  
(But your mark will be boosted up to your exam mark if that is higher.)
- You may use calculators and paper translation dictionaries.
- You may write notes and working on this paper, but make sure it is clear where your answers are.

### Questions

### Marks

1. Understanding Java

[17]

2. Event Driven Input

[10]

3. Defining Classes

[9]

4. File Processing

[9]

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 Java

[17 marks]

(a) [4 marks] Consider the following counting method:

```
public void counting(int max){  
    int num = 1;  
    int count = 0;  
    while (num < max){  
        count++;  
        Ul.println (count + " : " + num);  
        num = num * 2;  
    }  
    Ul.println ("final: " + num);  
}
```

max:

num:

count:

What will be printed if counting(16) is called?

(Question 1 continued on next page)

**(Question 1 continued)**

**(b)** [3 marks] Consider the following winner method:

```
public void winner(String x, String y, String z){
    if ( x.equals(y) ) {
        Ul.println ("VUW");
    }
    else if ( y.startsWith(z) ){
        Ul.println ("Canterbury");
    }
    else if ( z.contains(x) ) {
        Ul.println ("Otago");
    }
    else {
        Ul.println ("Auckland");
    }
}
```

(Refer to page 3 of the Java documentation for the meaning of `startsWith(...)` and `contains(...)`.)

What will be printed if `winner( "John", "John", "Jo" )` is called?

What will be printed if `winner( "Bob", "Bill","Bobby" )` is called?

What will be printed if `winner( "Will", "William", "Will" )` is called?

(Question 1 continued on next page)

**(Question 1 continued)**

**(c) [4 marks]** Suppose the file called `SomeText.txt` contains the following:

```
1003 laptop 1400
1009 disk 279
1021 laser 1899
1017 printer 630
```

What will the following `filePrint` method print out?  
Hint: keep track of the values of all the variables

```
public void filePrint (){
    try{
        Scanner scan = new Scanner (new File("SomeText.txt"));
        int max = 100;
        while ( scan.hasNext() ) {
            int x = scan.nextInt ();
            String token = scan.next();
            int y = scan.nextInt ();
            if (x > y) {
                max = y;
            }
            UI.println (max + " for " + token + " : " + x);
        }
        UI.println ("done: " + max);
        scan.close();
    }
    catch(IOException e){UI.println("Fail: " + e);}
}
```

max:	<input type="text"/>
x:	<input type="text"/>
token:	<input type="text"/>
y:	<input type="text"/>

(Question 1 continued on next page)

**(Question 1 continued)**

**(d)** [6 marks] Consider the Laptop class on the facing page.

What will the following fragment of code print out? (Note the variables carefully.)

```
Laptop lt1 = new Laptop("Windows", 8.0, 2);  
lt1 .upgrade(8.2);  
lt1 .report ();  
Laptop lt2 = new Laptop("MacOS", 10.7, 4);  
lt1 .expand(1);  
lt2 .expand(2);  
lt1 .report ();  
lt2 .report ();  
lt1 .upgrade(11.0);  
lt2 .report ();
```

lt1:

lt2:

(Question 1 continued on next page)

(Question 1 continued)

```
public class Laptop{

    // Fields
    private String opSys;
    private double version;
    private int memory;

    // Constructor
    public Laptop(String os, double ver, int mem){
        this.opSys = os;
        this.version = ver;
        this.memory = mem;
    }

    // Methods
    public void report(){
        Ul.println (this.opSys + " (" + this.version + ") " + this.memory + "GB");
    }

    public void upgrade(double vers){
        this.version = vers;
        Ul.println ("upgraded to " + this.version);
    }

    public void expand(int m){
        this.memory = this.memory + m;
        Ul.println ("memory - " + this.memory);
    }
}
```

## Question 2. Event-Driven Input

[10 marks]

The ImagePlacer program on the facing page should allow the user to draw a collection of images on the graphics pane, as shown below.

You are to complete the ImagePlacer program.

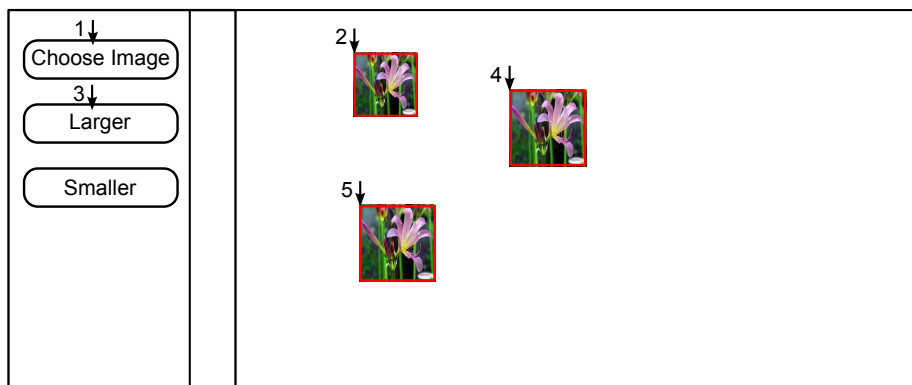
The Choose Image button should allow the user to select an image file with the UFileChooser.

If the user releases the mouse at a point on the graphics pane, then the program should draw the current image at the point, scaling both width and height to the current size. The program should draw a red border around the image. The program has a field containing the current size that the images will be drawn (initially 40).

The two buttons labeled Larger and Smaller should allow the user to change the current size, increasing or decreasing it by 10 units.

For full marks, your program should not crash if the user clicks the mouse in the graphics pane before they have chosen an image.

The diagram shows the result of the user choosing an image (1), then clicking at position (2) then clicking the the Larger button (3), then clicking at positions (4) and (5).



**Hint:** `Ul.drawImage("myhouse.jpg", 100, 200, 50, 60)` will draw the image in the file `myhouse.jpg` at position (100, 200) scaled to width 50 and height 60.

(Question 2 continued on next page)



(Question 2 continued)

```
public class ImagePlacer {
    private String imageName; // name of the image file
    private double size = 40; // size of images

    public ImagePlacer(){

    }

    public void buttonPerformed(String button){

    }

    public void mousePerformed(String action, double x, double y){

    }

    public static void main(String[] arguments){
        new ImagePlacer();
    }
}
```

### Question 3. Defining Classes

[9 marks]

For this question, you are to complete part of a program for a computer game involving a forest of magical mushrooms. The mushrooms are drawn on the screen as coloured circles. Initially, they are blue, and 20 units across. Every time the player touches a mushroom, it grows a bit larger. When it gets larger than a certain size ('activationSize'), it becomes "activated" and changes from blue to red. When it is activated, it affects the user in dangerous ways. A mushroom can be reset to its original size and then becomes unactivated.

Complete the Mushroom class on the facing page which defines Mushroom objects. You need to define

- Fields: to store any state information about the mushroom.
- The constructor: initialises the Mushroom. Parameters are the x and y coordinates of the *centre* of the Mushroom.
- The grow method: Parameter is the amount to grow by. Increases the size of the Mushroom by the amount.
- The reset method: No parameters. Resets the Mushroom to the initial size.
- The isActivated method: No parameters. Returns a boolean value that is true if the mushroom is activated and false otherwise.
- The draw method: No parameters. Draws the mushroom as a coloured circle of the appropriate size; red if activated and blue otherwise.

Note, the constructor, the grow method and the reset method do not need to draw the Mushroom.

As an example of using the Mushroom class, the testMushroom method below creates a Mushroom object, centered at position (100,250), makes it grow in steps of 5 units until it is activated, and then resets it.

```
public static void testMushroom(){
    Mushroom mush = new Mushroom(100,250);
    mush.draw();
    while (! mush.isActivated() ){
        UI.clearGraphics();
        mush.grow(5);
        mush.draw();
        UI.sleep(100);
    }
    UI.clearGraphics();
    mush.reset();
    mush.draw();
}
```

(Question 3 continued)

```
public class Mushroom{
    private static final double initialSize = 20;
    private static final double activationSize = 55;

    public Mushroom(double x, double y){

    }

    public void grow(double step){

    }

    public void reset(){

    }

    public boolean isActivated(){

    }

    public void draw(){

    }

}
```

#### Question 4. File Processing

[9 marks]

Suppose the file `equipment.txt` contains information about the electronic testing equipment owned by a university. Each line contains information about one item: the manufacturer, the price, the serial number, and the type of equipment.

For example, part of the file might be:

```
Hewlett Packard 14000 BN3058323 Oscilloscope
Bosch Corp 2850 TRP3098572 Range Finder
Agilent Technologies 9725 38532354QR Signal Generator
Leupold 1350 S25348374 Range Finder
Agilent Technologies 641500 23432YD234 NMR Spectrometer
Dimex Measurement Corp 260 2398479823 Range Finder
```

The `listItems` method on the facing page has two parameters — a type of equipment and a maximum price. Complete the `listItems` method so that it reads the “`equipment.txt`” file and prints out the serial number, manufacturer, and price of each item of the specified type whose price is at most the specified maximum price. Finally, it prints out the number of matching items.

For example, if the file contained just the data above, then calling

```
listItems("Range Finder", 2000);
```

should print out

```
Range Finder: costing $2000 or less:
  S25348374 Leupold $1350
  2398479823 Dimex Measurement Corp $260
Total: 2
```

(Question 4 continued on next page)

(Question 4 continued)

```
public void listItems (String type, int maxPrice){
    try{
        Ul.println (type + ": costing $" + maxPrice + " or less:");

        scan.close();
    }
    catch(IOException e){Ul.println("File Failure: " + e);}
}
```

\*\*\*\*\*

**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.