

EXAMINATIONS – 2019

TRIMESTER 2

<p>COMP 102 INTRODUCTION TO COMPUTER PROGRAM DESIGN</p>

Time Allowed: TWO HOURS

OPEN BOOK

Permitted materials: Silent non-programmable calculators or silent programmable calculators with their memories cleared are permitted in this examination.

Printed foreign language–English dictionaries are permitted.

You may access the Java Documentation.

You may not access any other websites or online help of any kind.

Instructions:

Attempt ALL Questions.

The exam will be marked out of 120 marks.

Write your answers in BlueJ, following the instructions on page 2.

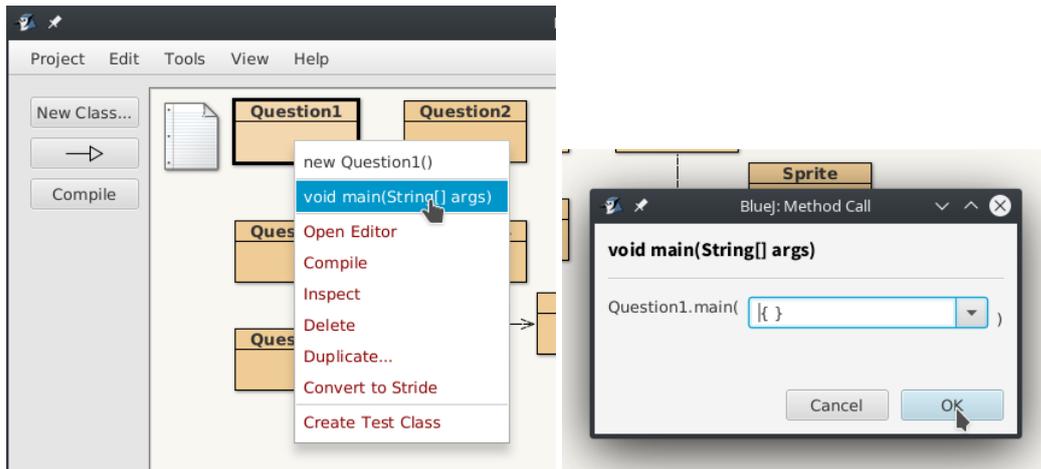
You may write notes and working on this paper, but make sure your answers are clear.

Questions:

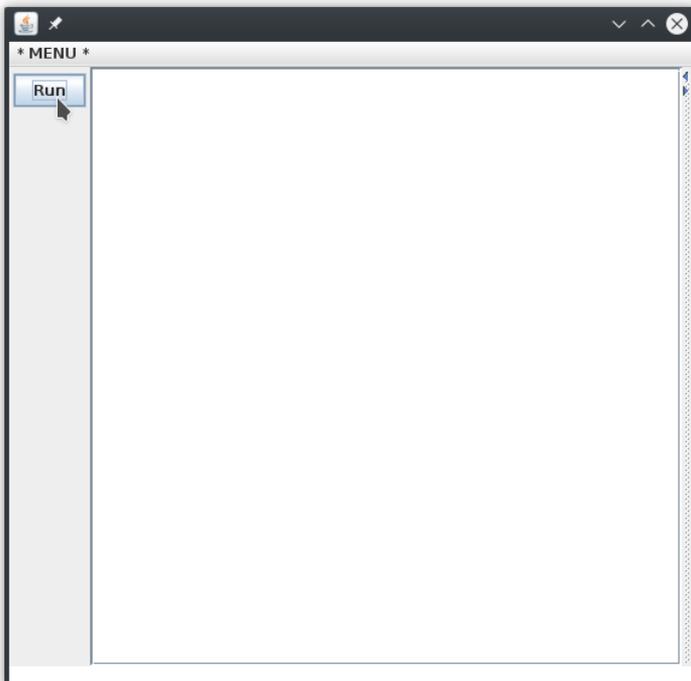
- | | |
|--------------------------|------|
| 1. Basic Java | [27] |
| 2. Design a class | [15] |
| 3. Files | [23] |
| 4. ArrayLists of Objects | [32] |
| 5. 2D Arrays | [23] |

Running the code:

1. Download the COMP102-final-exam.zip file from <https://ecs.victoria.ac.nz/Courses/COMP102.2019T2/COMP102-final-exam>.
2. Unzip the file, and open the project in BlueJ.
3. Each question has been given its own class. To run them, right-click and select 'void main(String[] args)'. Hit 'OK' when it prompts you for arguments.



4. The UI that pops up will have a 'Run' button that will execute the code for the question, once you've written it.



Question 1. Basic Java**[27 marks]****(a) [4 marks] Writing methods.**

Complete the method `myMethod` in `Program1` so that the following method prints out the example given below. `myMethod` should compute the sum of its arguments.

```
public void printStuff (){
    Ul.println ("printing stuff");
    int x = 5;
    int y = 9;
    Ul.println ("x=" + x + " y=" + y);
    int z = x * y;
    Ul.println ("z=" + z);
    z = this.myMethod(x, y);
    Ul.println ("z=" + z);
}

public int myMethod(int m, int n){
    /*# YOUR CODE HERE */
}
```

Output:

```
printing stuff
x=5 y=9
z=45
z=14
```

(Question 1 continued)

Consider the following Circle class, specifying objects with four fields, a constructor and four methods.

```
class Circle {
    private Color color;
    private double size;
    private double x,y;

    public Circle(double x, double y, double size, Color color){
        this.x = x;
        this.y = y;
        this.size = size;
        this.color = color;
    }

    public void grow(){
        this.erase();
        this.size = this.size + 5;
        this.draw();
    }

    public void shrink(double amount){
        this.erase();
        this.size = this.size - amount;
        this.draw();
    }

    public void draw(){
        UI.setColor(this.color);
        UI.fillOval(x-size/2, y-size/2, size, size);
    }

    public void erase(){
        UI.eraseOval(x-size/2, y-size/2, size, size);
    }
}
```

(Question 1 continued)

(b) [6 marks] Calling methods on objects.

Given the Circle class on the facing page, complete the useCircle() method in Program1 so that it:

- makes a green circle at (100,150) with a size of 20
- makes a red circle at (150, 200) with a size of 15
- draws both circles on the screen
- makes the first circle grow 3 times
- makes second circle grow 2 times
- shrinks the first circle 10 pixels
- shrinks the second circle 2 pixels

(c) [8 marks] Arrays.

Complete the sumArray() method in Program1 so that it adds up all the numbers in the nums array (when testing your code, check that the answer is 27).

(d) [9 marks] ArrayLists.

Complete the countNegatives() method in Program1 so that it counts and prints the number of negative values that appear in the ArrayList of Doubles stored in nums.

For example, if the user enters "-1 8.8 -2.1 0 9 2 5", it should print "There were 2 negative numbers."

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

Suppose you are writing a program to simulate balls falling down a waterfall from the top of the screen to the bottom.

Complete the Ball class in BlueJ to specify Ball objects that are used by the simulation in Program2.

- The fields, and the draw() and erase() method have been defined for you.
- The constructor should create a Ball object with the specified size, speed and color. It should call the draw method to show the ball at the default start position.
- The move() method should move the ball downwards according to its speed. It should not redraw it.
- The getY() method should return the y position of the ball.
- The setSpeed(...) method should update the speed of the ball.

Question 3. Files

[23 marks]

A file is shown below. Each line contains the definition of a Shape (defined in the Shape class): the type, the x position, the y position, the size, whether it is filled or not, and the color (as 3 integers, representing the Red, Green, and Blue values of the color).

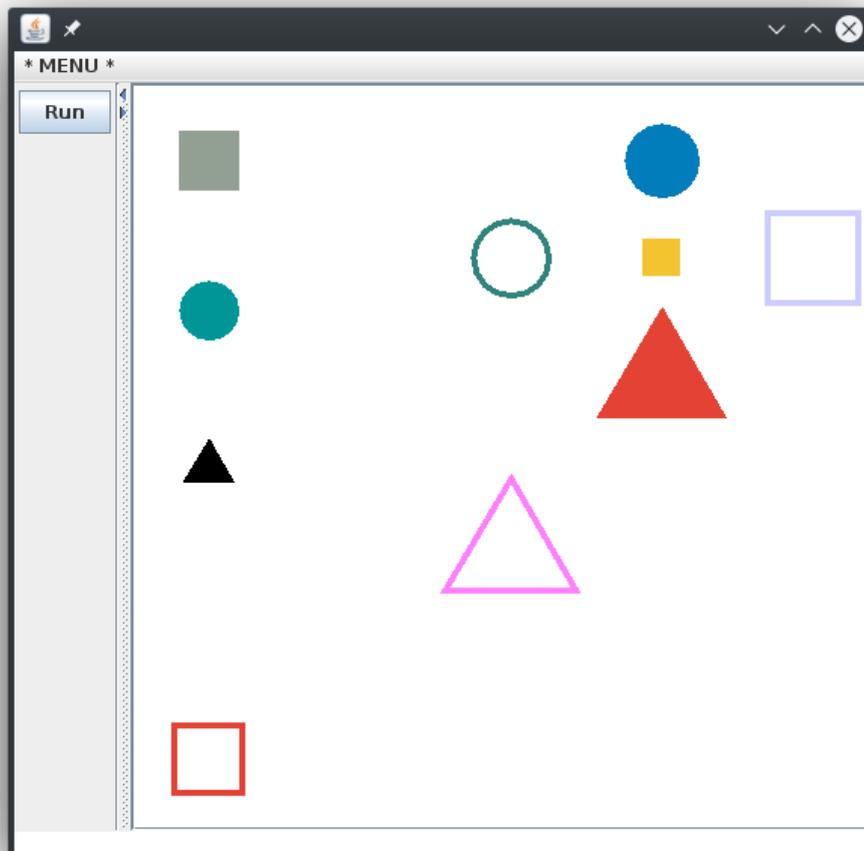
```
square 50 50 40 true 145 160 146
circle 200 50 50 true 0 125 186
triangle 200 200 75 true 227 66 52
square 250 100 90 false 204 204 255
circle 100 100 50 false 47 132 124
triangle 100 400 75 false 255 128 255
...
```

Complete the `run()` and `processLine(...)` methods in Program3 so that they will read the file, create a Shape object for each line, and draw the shape on the screen.

The `run()` method should use a scanner to read each line, and then call `processLine(...)` to create a shape.

`processLine(...)` should construct a Shape object from the string provided as its argument.

The name of the data file is `data.txt`. The output should look like the image below:



THIS PAGE INTENTIONALLY LEFT BLANK

Question 4. ArrayLists of Objects

[32 marks]

This question concerns a program for scheduling meetings. Answer this question in the Program4 class.

The program has two classes: Program4 and a Meeting class.

Program4 has a schedule field which contains an ArrayList of Meeting objects.

The Meeting class below defines Meeting objects, which store the date and hour of the meeting.

The date is a string, such as "14 August".

The hour is stored as an int, and should have a value of 9, 10, 11, 12, 1, 2, 3, or 4 (meetings can only be scheduled during work hours).

```
class Meeting {
    private String date;
    private int hour;

    public Meeting(String d, int h) {
        this.date = d;
        this.hour = h;
    }

    public String getDate(){
        return this.date;
    }

    public int getHour(){
        return this.hour;
    }
}
```

(a) [15 marks] Complete the searchByMonth(...) method that will search the schedule to print the information for all meetings that occur in the month specified by the parameter. For example, searchByDate(" August") should print the date and hour for every meeting that occurs in August, e.g.

```
14 August @ 2
8 August @ 11
9 August @ 12
```

If no matching meetings are found, this method should display a "no meetings" message.

(b) [17 marks] Complete the addMeeting(...) method that will create a meeting with the date and hour specified in the parameters, and then add the meeting to the schedule **if and only if** it does not conflict with an existing meeting.

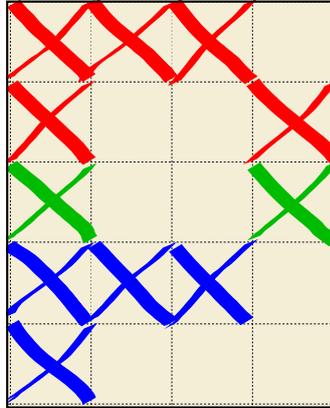
A meeting conflicts if it occurs on the same date, at the same hour.

The method should return true if the meeting was successfully added, or false otherwise.

Question 5. 2D Arrays

[23 marks]

Cross-stitch is a form of embroidery that uses cross-shaped stitches in a grid pattern. A pattern looks very much like a picture, with each pixel being one stitch. For example, this is a simple pattern for the letter "P", in three colors:



The pattern field is declared to hold a 2D array as follows. A color value indicates that the stitch should be done with that colour of thread. A null value indicates that there is no stitch in that location.

```
private Color[ ][ ] pattern = new double[ ][ ]{{ Color.RED, Color.RED, Color.RED, null},
                                                { Color.RED, null, null, Color.RED},
                                                { Color.GREEN, null, null, Color.GREEN},
                                                { Color.BLUE, Color.BLUE, Color.BLUE, null},
                                                { Color.BLUE, null, null, null }};
```

(a) **[8 marks]** Complete the findStitchesPerRow() method in Program5 that will find and print the number of stitches on each row.

For example, if the 2D array contains the data given above, findStitchesPerRow() should print out the following:

- 0: 3
- 1: 2
- 2: 2
- 3: 3
- 4: 1

Your method should still work whatever the size of the array, assuming that every row of the array contains at least one column.

(Question 5 continued on next page)

(Question 5 continued)

(b) [15 marks] Complete the `countSpecialStitches(...)` method in `Program5` that will find and print the total number of stitches that match the color provided in the argument **and** share a colour with the stitch to their right.

For example, if the 2D array contains the data given above, `countSpecialStitches(Color.RED)` should print out the following:

2 stitches that match the color and the stitch to their right

Your method should still work whatever the size of the array, assuming that every row of the array contains at least one column.
