

Name:

ID Number:

Signature:

COMP102: Test Model Solutions

13 September, 2004

Instructions

- Time allowed: $1\frac{1}{2}$ hours.
- Answer **all** the questions.
- There are 100 marks in total.
- Write your answers in the boxes in this test paper and hand in all sheets.
- If you think some question is unclear, ask for clarification.
- This test will contribute 25% of your final grade.
- Numeric keypad calculators and non-electronic dictionaries are permitted.

Questions

Marks

1. Understanding Java

[34]

2. Programming with conditionals

[12]

3. Programming with loops

[12]

4. Programming with objects

[20]

5. Programming with arrays

[22]

TOTAL:

Question 1. Understanding Java programs

[34 marks]

For each of the following programs (a)-(f), show the output produced when the program is run.

(a) [5 marks]

```
public class Test1 {
    public static void main(String[] args) {
        int x = 12;
        int y = 10;
        int z = 5;
        System.out.println("This is " + (x + y + z));
        System.out.println("That is " + x + " + " + y + " + " + z);
        System.out.println(x + y * z);
        System.out.println(x / z);
        System.out.println(Math.max(Math.min(x,y), z));
    }
}
```

This is 27
That is 12 + 10 + 5
62
2
10

(b) [10 marks]

```
public class Test2 {
    public static void main(String[] args) {
        String a = "Computers are fun";
        String b = "Say it again!";
        String c = "9876543210";

        int i = a.indexOf(" ");
        int j = b.indexOf(" ");
        int k = c.indexOf("1");
        System.out.println(i);
        System.out.println(j);
        System.out.println(k);
        System.out.println(a.substring(0, j+1));
        System.out.println(c.substring(k) + c.substring(k-1,k));
        System.out.print(b.charAt(4));
        System.out.println(a.charAt(8));
        System.out.println(a.substring(a.length()-3));
        if ( a.substring(i, i+2).equals(b.substring(j+3, j+5)) )
            System.out.println("Yes");
        else
            System.out.println("No");
    }
}
```

```
9
3
8
Comp
102
is
fun
Yes
```

(c) [5 marks]

```
public class Test3 {
    public static void main(String[] args) {
        int a = 26;
        int b = 5;
        int c = 12;
        String r = "";
        String s = "";
        if ( 2*c <= a-b )
            r = "red";
        else
            r = "blue";
        System.out.println(r);
        if ( b < c && c < a )
            s = "car";
        else
            s = "van";
        System.out.println(s);
        if ( r == null || s == null )
            System.out.println("Error!");
        else
            if ( r.length() == s.length() )
                System.out.println(r + " " + s);
            else
                System.out.println(s + " " + r);
    }
}
```

```
blue
car
car blue
```

(d) [4 marks]

```
public class Test4 {  
    public static void main(String[] args) {  
        String s = "abcdefghijklmnopqrstuvwxy";  
        String r = "";  
        int k = 4;  
        while ( k < s.length() ) {  
            r = r + s.charAt(k);  
            System.out.println(k + " : " + r);  
            k = k+6;  
        }  
    }  
}
```

```
4: e  
10: ek  
16: ekq  
22: ekqw
```

(e) [4 marks]

```
public class Test5 {
    public static void main(String[] args) {
        Tally t = new Tally();
        int[] a = { 5, 3, 0, 8, -1, -7, 50, 0, -1, 0, 337 };
        for (int i = 0; i < a.length; i++)
            if ( a[i] == 0 )
                t.add();
        System.out.println("Tally is " + t.get());
    }
}
```

```
class Tally {
    private int k = 0;
    public void add() {
        k++;
    }
    public int get() {
        return k;
    }
}
```

Tally is 3

(f) [6 marks]

```
public class Test6 {
    public static void main(String[] args) {
        Pair u = new Pair(1, 2);
        Pair v = new Pair(u.getx(), u.gety() );
        if ( u == v )
            System.out.println("Yes");
        else
            System.out.println("No");
        if ( u.equals(v) )
            System.out.println("Yes");
        else
            System.out.println("No");
    }
}
```

```
class Pair {
    private int x;
    private int y;
    public Pair(int xx, int yy) {
        x = xx;
        y = yy;
        System.out.println( toString() );
    }
    public int getx() {
        return x;
    }
    public int gety() {
        return y;
    }
    public boolean equals(Pair that) {
        return (x == that.x) && (y == that.y);
    }
    public String toString() {
        return x + " " + y;
    }
}
```

1 2
1 2
No
Yes

Question 2. Programming with conditionals

[12 marks]

Workers in a factory that makes garden tables are paid according to how many tables they produce each week. If they produce between 10 and 30 tables, they are paid the standard rate of \$10 per table; if they produce more than 30 tables, they are paid a bonus rate of \$15 per table; and if they produce less than 10 tables, they are paid a penalty rate of \$5 per table.

Below is the outline of a program to compute the amount to be paid to a worker. The program should read a worker's name and the number of tables they produced, and print a message showing the worker's name and the amount they are to be paid. In addition, if the worker produced less than 5 tables or more than 50, the program should print a message asking the worker to report to his/her supervisor (to discuss their performance).

Complete the program, by adding the code required to compute the worker's pay and print the required output(s).

Note that the program is not required to check the validity of the inputs read.

```
import javax.swing.*;

public class ComputePay {

    public static void main(String[] args) {

        // Read the worker's name
        String name = JOptionPane.showInputDialog("Enter name");

        // Read the number of table produced as a string and turn it into an int
        String s = JOptionPane.showInputDialog("Enter number of tables produced");
        int count = Integer.parseInt(s);

        int rate = 10;
        if ( count < 10 ) rate = 5;
        else if ( count > 30 ) rate = 15;

        System.out.println("Pay for " + name + " is " + count*rate);

        if ( count < 5 || count > 50 )
            System.out.println("Please see your supervisor");

    }
}
```

Question 3. Programming with loops

[12 marks]

Complete the following program so that it reads an integer n , and prints a triangle of asterisks. The triangle consists of n lines. The first line has one asterisk, the second line has two asterisks, and so on until the n th line which has n asterisks. For example, if the input is 5, the program should print:

```
*
**
***
****
*****
```

You may assume that the input entered is a positive integer.

```
import javax.swing.*;

public class DrawTriangle {

    public static void main(String args[]) {

        // Read the triangle size as a string and turn it into an int
        // Assumed to be a positive integer
        String s = JOptionPane.showInputDialog("Enter size");
        int size = Integer.parseInt(s);
```

```
// Print the triangle

    for (int i = 1; i <= size; i++) {
        for (int j = 1; j <= i; j++) {
            System.out.print("*");
        }
        System.out.println();
    }
    System.exit(0);
```

```
    }
}
```

Question 4. Programming with objects

[20 marks]

The following is a simple program for constructing a shopping list:

```
public class Shopper {
    public static void main(String[] args) {
        ShoppingList s = new ShoppingList();
        s.addItem("bread", 1, 2.40);
        s.addItem("milk", 2, 3.60);
        s.addItem("paper", 1, 1.30);
        s.showList();
    }
}
class ShoppingList {
    private String slist = "";
    public ShoppingList() {
    }
    public void addItem(String item, int qty, double unitPrice) {
        double price = qty * unitPrice;
        slist = slist + item + " " + qty + " @ " + unitPrice + " = " +
            price + "\n";
    }
    public void showList() {
        System.out.println(slist);
    }
}
```

(a) [10 marks] What output will this program produce?

```
bread 1 @ 2.4 = 2.4
milk 2 @ 3.6 = 7.2
paper 1 @ 1.3 = 1.3
```

(b) [10 marks] Suppose you want a call on `showList` to also print the total cost of all the items on the shopping list. Show the changes you would need to make on the copy of the `ShoppingList` class given in the answer box below. (You should not change the `Shopper` class.)

```
class ShoppingList {

    private String slist = "";

    private double total = 0;           // Declare a field to hold the total

    public ShoppingList() {

    }

    public void addItem(String item, int qty, double unitPrice) {

        double price = qty * unitPrice;

        slist = slist + item + " " + qty + " @ " + unitPrice + " = " +
            price + "\n";

        total = total + price;         // Add price to total

    }

    public void showList() {

        System.out.println(slist + "" + total);    // Print total

    }

}
```

Question 5. Programming with arrays

[22 marks]

(a) [2 marks] Write a Java statement to declare and create an array **a** of 10 characters.

```
int[] a = new int[10];
```

(b) [2 marks] Write a Java statement to declare and create an array **b** of 5 integers, and initialise it so that it contains the values 3, 5, 7, 1, and 12, in that order.

```
int[] b = { 3, 5, 7, 1, 12 };
```

(c) [2 marks] Write a Java statement to print the first and last elements of an array **c**, on the same line, assuming that **c** contains a valid array value with at least one element.

```
System.out.println(c[0] + " " + c[c.length-1]);
```

(d) [8 marks] Complete the following method, so that it takes an array of integers `nums` as an arguments and returns the largest value in `nums`. Assume that `nums` contains a valid array value with at least one element.

```
// Find the largest value in nums
public int maxNum(int[] nums) {
```

```
    int m = nums[0];
    for (int i = 1; i < nums.length; i++)
        if ( nums[i] > m ) m = nums[i];
    return m;
```

```
}
```

Note that initialising `m` to 0 will not give the correct result if all elements on `nums` are negative!

(e) [8 marks] Complete the following method, so that it takes two array of integers **a** and **b** as an arguments, and returns an array containing all the elements of **a**, followed by all the elements of **b**. For example, if **a** is an array of size 5, containing the values 3, 5, 1, 9, 6, and **b** is an array of size 4, containing the values 5, 5, 1, 2, the result should be an array of size 9, containing the values 3, 5, 1, 9, 6, 5, 5, 1, 2. You may assume that **a** and **b** contain valid array values.

```
// Join two arrays together  
public int[] joinArrays(int[] a, int[] b) {
```

```
    int[] c = new int[a.length + b.length];  
    for (int i = 0; i < a.length; i++)  
        c[i] = a[i];  
    for (int i = 0; i < b.length; i++)  
        c[i+a.length] = b[i];  
    return c;
```

```
}
```

```
*****
```