

Name:

ID Number:

Signature:

COMP102: Test Model Solutions

22 April, 2002

Instructions

- Time allowed: **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

[38]

2. Programs with conditionals

[12]

3. Programs with loops

[10]

4. Programs with objects

[20]

5. Programs with arrays

[20]

TOTAL:

Question 1. Understanding Java

[38 marks]

For each of the following programs, show the output produced when the program is run.

(a) [5 marks]

```
public class Test1 {
    public static void main(String[] args) {
        int x, y, z;
        x = 2;
        y = 5;
        z = 10;
        System.out.println(x * y + z);
        System.out.println(y / x);
        System.out.println("Sum1 is " + (x + y + z));
        System.out.println("Sum2 is " + x + " + " + y + " + " + z);
        System.out.println(Math.min(Math.min(x,y),z));
    }
}
```

```
20
2
Sum1 is 17
Sum2 is 2 + 5 + 10
2
```

(b) [5 marks]

```
public class Test2 {
    public static void main(String[] args) {
        String s = "VUW Library";
        System.out.println(s.length());
        System.out.println(s.charAt(2));
        System.out.println(s.substring(4, 9));
        System.out.println(s.indexOf(" "));
        System.out.println(s.substring(s.length()-2) + s.substring(0,s.indexOf(" ")));
    }
}
```

```
11
W
Libra
3
ryVUW
```

(c) [4 marks]

```
public class Test3 {
    public static void main(String[] args) {
        int a, b;
        a = 6;
        b = 15;
        if (b < 2*a)
            System.out.print("red ");
        else
            System.out.print("blue ");
        System.out.println("grass");
    }
}
```

blue grass

(d) [6 marks]

```
public class Test4 {
    public static void main(String[] args) {
        String s = "Alexander Bean";
        String t = "Bean, Alexander";
        System.out.println(munge(s));
        System.out.println(munge(t));
    }

    public static String munge(String s) {
        int i = s.indexOf(",");
        int j = s.indexOf(" ");
        if ( i == -1 )
            return s.charAt(0) + " " + s.substring(j+1);
        else
            return s.charAt(j+1) + " " + s.substring(0, i);
    }
}
```

A Bean
A Bean

(e) [10 marks]

```

public class Test5 {
    public static void main(String[] args) {
        int x, y, z;
        x = 1;
        y = 1;
        z = 0;
        while ( y < 15 ) {
            System.out.println(x + " " + y + " " + z);
            x = x+1;
            y = 2*y;
            z = z + y + 1;
        }
        System.out.println(x + " " + y + " " + z);
    }
}

```

```

1 1 0
2 2 3
3 4 8
4 8 17
5 16 34

```

(f) [8 marks]

```

public class Test6 {
    public static void main(String[] args) {
        int n = 3;
        int k = 1;
        int i = 1;
        while ( i <= 2*n-1 ) {
            int j = 1;
            while ( j <= Math.min(i, 2*n-i) ) {
                System.out.print(k + " ");
                j = j+1;
                k = k+1;
            }
            i = i+1;
            System.out.println();
        }
    }
}

```

```

1
2 3
4 5 6
7 8
9

```

Question 2. Programs with conditionals

[12 marks]

Students sitting an exam receive a *Pass With Distinction* if they get a mark of 85 or more, a *Pass With Merit* if their mark is in the range 70 to 84, a plain *Pass* if their mark is in the range 50 to 69, and *Fail* if their mark is below 50.

Below is the outline of a program to read a student's name and mark, and print out the candidate's name and result. Complete the program, by adding the code required to set `res` to a string indicating the result for this student, as described above. You should assume that `s` is a string of digits, corresponding to a whole number between 0 and 100, inclusive.

```
import javax.swing.*;

public class ExamResults {

    public static void main(String args[]) {

        String name;
        int mark;
        String res;

        name = JOptionPane.showInputDialog("Enter name");

        // Read a mark and turn it into a number
        String s = JOptionPane.showInputDialog("Enter mark");
        mark = Integer.parseInt(s);
```

```
// Set res to a string indicating the result for this student
```

```
if ( mark >= 85 ) res = "Pass with Distinction";
else if ( mark >= 70 ) res = "Pass with Merit";
else if ( mark >= 50 ) res = "Pass";
else res = "Fail";
```

OR:

```
if ( mark < 50 ) res = "Fail";
else if ( mark < 70 ) res = "Pass";
else if ( mark < 85 ) res = "Pass with Merit";
else res = "Pass with Distinction";
```

```
JOptionPane.showMessageDialog(null,
    "Result for " + name + " is " + res);
}
}
```

Question 3. Programs with loops

[10 marks]

A simple way to display information about appointments (such as meetings, lectures, tutorials, etc.) is to print a line of characters for each appointment, with asterisks representing the duration of the appointment and spaces elsewhere. This will allow you to easily see whether two appointments overlap, how long there is between the end of one appointment and the start of another, etc. For example, if you have one appointment from 10:00 till 12:00 and another from 11:00 till 15:00, the display might look like this:

```

9   10  11  12  13  14  15  16  17  18
      *****
          *****

```

In producing this display, we assume that all appointments fall between 9:00 (9am) and 18:00 (6pm). Appointments are considered to start on the hour and last for a whole number of hours. Each hour is represented by four characters, so that there is enough space to print the numbers along the top.

Opposite is the outline of a program to read the start and end times of two appointments, and produce the kind of display described above. The program uses a method called `timeString` which returns a string of characters showing the duration of a given appointment. The first parameter gives the hour at which the appointment begins, and the second parameter gives the hour at which the appointment ends.

Complete the implementation of `timeString`, so that it return a string consisting of four spaces for each hour between 9:00 and the start of the appointment followed by four asterisks for each hour from the beginning to the end of the appointment.

```

import javax.swing.*;

public class ShowTimes {

    public static void main (String[] args) {

        int start1 = getInt("Enter start time of first appointment");
        int end1 = getInt("Enter end time of first appointment");
        int start2 = getInt("Enter start time of second appointment");
        int end2 = getInt("Enter end time of second appointment");

        System.out.println("9  10  11  12  13  14  15  16  17  18");

        System.out.println(timeString(start1, end1));

        System.out.println(timeString(start2, end2));

    }

    // Make a string depicting the duration of an appointment.
    public static String timeString(int startHour, int endHour) {

        int h;
        String s = "";
        for (h = 9; h < startHour; h++)
            s = s + "  ";
        for (h = startHour; h < endHour; h++)
            s = s + "****";
        return s;

    }

    // Read an integer
    public static int getInt(String msg) {
        String s = JOptionPane.showInputDialog(msg);
        return Integer.parseInt(s);
    }
}

```

Question 4. Programs with objects

[20 marks]

Consider the following class `Point` to represent a point in two-dimensional space.

```
class Point {  
    private int x, y;  
  
    public Point(int xx, int yy) {  
        x = xx; y = yy;  
    }  
  
    public int getX() {  
        return x;  
    }  
  
    public int getY() {  
        return y;  
    }  
  
    public void move(int dx, int dy) {  
        x = x + dx; y = y + dy;  
    }  
  
    public void show() {  
        System.out.println("(" + x + ", " + y + ")");  
    }  
}
```

(a) [6 marks] What output will this program produce?

```
public class Points {  
  
    public static void main(String[] args) {  
        Point p = new Point(10, 30);  
        Point q = new Point(40, 20);  
        p.show();  
        q.show();  
        p.move(q.getX(), q.getY());  
        p.show();  
    }  
}
```

```
(10,30)  
(40,20)  
(50,50)
```

(b) [8 marks] A point is on the x -axis if its y coordinate is zero, above the x -axis if its y coordinate is positive, and below the x -axis if its y coordinate is negative.

Complete the following method (to added to the `Point` class) to determine whether a point is on, above or below the x -axis. The method should return 0 if it is on the x -axis, 1 if it is above the x -axis, and -1 if the point is below the x -axis.

```
public int Alignment() {  
  
    if ( y < 0 ) return -1;  
    else if ( y == 0 ) return 0;  
    else return 1;  
  
}
```

(c) [6 marks] One point is to the left of another point if its x coordinate is less than that of the other point.

Complete the following method (to be added to the `Point` class) to determine whether a point is to the left of a second point which is passed as a parameter.

```
public boolean leftOf (Point that) {  
  
    return x < that.x;  
  
OR:  
  
    return x < that.getx();  
  
}
```

Question 5. Programs with arrays

[20 marks]

We have a class consisting of a data field and three methods: `print1()`, `print2()` and `print3()`. The data field is declared and created as an array with 5 numbers.

```
private int[ ] nums = {45, 16, 25, 80, 50};
```

The array looks like this:

nums[0]	nums[1]	nums[2]	nums[3]	nums[4]
45	16	25	80	50

(a) [7 marks] What will the following method print out?

```
public void print1() {
    int i = 1;
    System.out.println(nums[2 * i]);
    int j = 2 * nums[i];
    System.out.println(j);
    for (int k = 4; k >= 0; k--)
        System.out.println(nums[k]);
}
```

25
32
50
80
25
16
45

(b) [5 marks] What will the following method print out?

```
public void print2() {
    for(int i=0; i < 5; i++){
        if (nums[i] >= 60)
            nums[i] = nums[i]-5;
        else if (nums[i] >= 30)
            nums[i] = 40;
        else
            nums[i] = nums[i]+5;
    }
    for(int i=0; i < 5; i++){
        System.out.println(nums[i]);
    }
}
```

40
21
30
75
40

(c) [8 marks] Write the method `print3()`. It should print “A” for numbers above 30 and print “B” for numbers below or equal to 30.

Suppose the array holds 45, 16, 25, 80, 50. It should print

A
B
B
A
A

```
public void print3() {  
  
    for(int i=0; i < 5; i++){  
        if( nums[i] > 30 )  
            System.out.println("A");  
        else  
            System.out.println("B");  
    }  
  
}
```
