

Name:

Usercode:.....

ID Number:.....

Signature:.....

COMP102: Test

5 May, 2003

Instructions

- Time allowed: **2 hours**.
- Answer **all** the questions.
- There are 120 marks in total.
- Write your answers in the boxes in this test paper and hand in all sheets.
- Use extra pages if necessary and write the question number clearly.
- If you think a question is unclear, ask for clarification.
- This test will contribute 25% to your final grade.
- Numeric keypad calculators and non-electronic dictionaries are permitted.

Questions

Marks

1. Understanding Java

[23]

2. Understanding simple programs

[27]

3. Programs with conditionals and loops

[25]

4. Programs with objects

[20]

5. Programs with arrays

[25]

TOTAL:

Question 1. Understanding Java

[23 marks]

Consider the following Java program:

```
import javax.swing.*;

public class StudentData {

    public static void main (String[] args) {

        Student s1 = new Student();

        Student s2 = new Student();

        String msg = s1.getResult();

        JOptionPane.showMessageDialog(null, msg);

        msg = s2.getResult();

        JOptionPane.showMessageDialog(null, msg);
    }
}

class Student {

    private String name;
    private double mark;

    public Student () {

        String s = JOptionPane.showInputDialog("Enter name and mark");

        int p = s.indexOf(" ");

        name = s.substring(0, p);

        String sMark = s.substring(p+1);

        mark = Double.parseDouble(sMark.trim());
    }

    public String getResult () {

        String message;

        if ( mark >= 50 ) {

            message = name + " has passed";

        } else {

            message = name + " has failed";

        }

        return message;
    }
}
```

(a) [4 marks] Write down the names (identifiers) of the classes declared in the program.

(b) [3 marks] Write down the names of the methods (including constructors) declared in the program.

(c) [4 marks] Write down the names of the `String` methods called in the program. (`String` methods are the methods declared in the `String` class)

(d) [4 marks] Write down the names of the data fields declared in the program.

(e) [4 marks] Write down the names of the local variables declared in the program.

(f) [4 marks] Write down the names of all the data types used in the program.

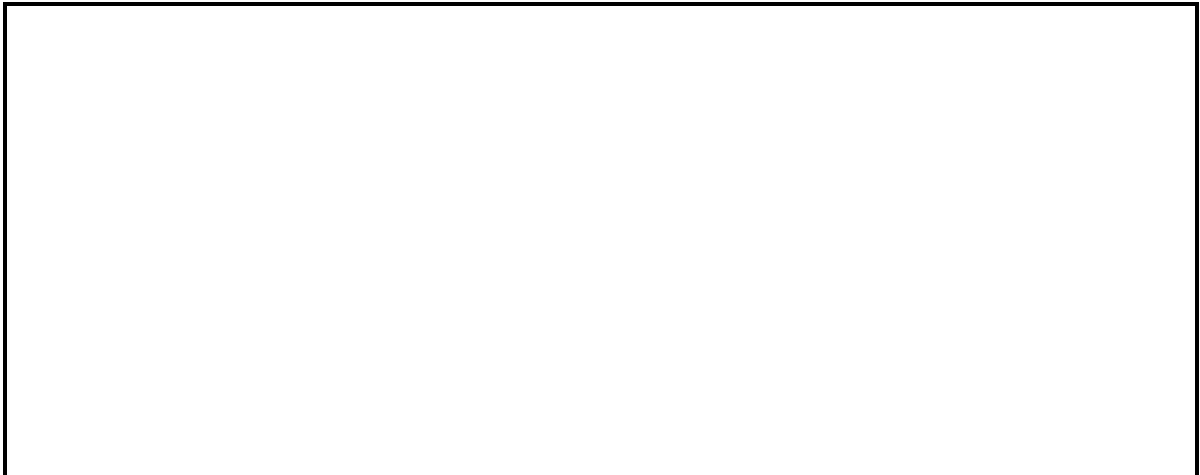
Question 2. Understanding simple programs

[27 marks]

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


(a) [6 marks]

```
public class Test1 {
    public static void main(String[] args) {
        String s = "Hello World!";
        System.out.println(s.charAt(0));
        System.out.println(s.substring(6));
        System.out.println(s.substring(2, 5));
        System.out.println(s.indexOf(" "));
        System.out.println(s.length());
        System.out.println(s.substring(s.length()-1));
    }
}
```



(b) [4 marks]

```
public class Test2 {
    public static void main(String[] args) {
        int a = 2;
        int b = 10;
        if ((a >= 0) && (a <= b))
            System.out.println(a);
        else
            System.out.println(b);
        System.out.println("***");
    }
}
```



(c) [5 marks]


```
public class Test3 {
    public static void main(String[] args) {
        int i = 8;
        while( i > 0) {
            i = i - 2;
            System.out.println(i);
        }
        System.out.println(i);
    }
}
```



(d) [4 marks]

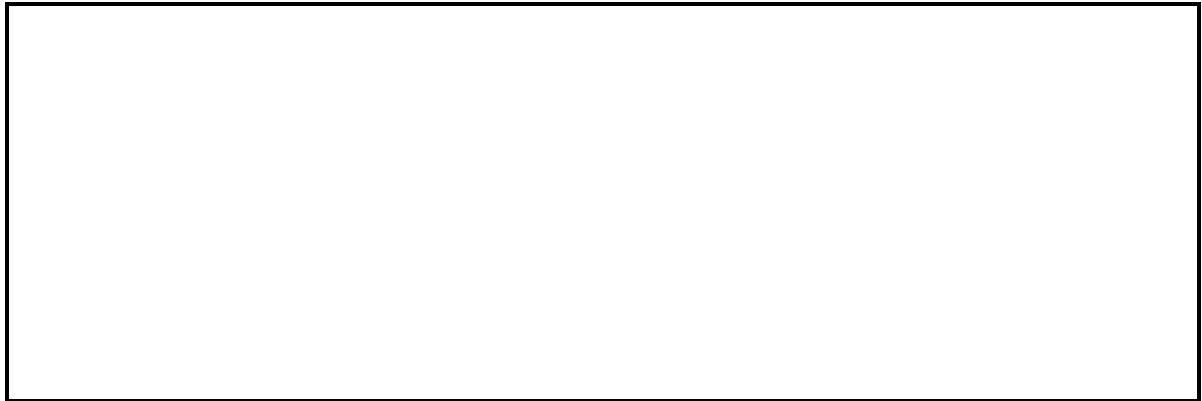
```
public class Test4 {
    public static void main(String[] args) {
        int x, y, z;
        x = 1;
        y = 1;
        z = 0;
        if ( x > y)
            System.out.println("Red");
        else if (x < y)
            System.out.println("Green");
        else if (z > y)
            System.out.println("Yellow");
        else
            System.out.println("White");

        System.out.println("***");
    }
}
```



(e) [8 marks]

```
public class Test5 {  
    public static void main(String[] args) {  
        int n = 3;  
        int k = 1;  
  
        for(int i = 0; i <= n; i++){  
  
            for(int j = 0; j <= n; j++) {  
                System.out.print(k + " ");  
                k = k+1;  
            }  
            System.out.println();  
        }  
    }  
}
```



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 3. Programs with conditionals and loops

[25 marks]

(a) [15 marks] Student data

Students sitting an exam receive an *A* if they get a mark of 75 or more, a *B* if their mark is in the range 65 to 74, a *C* if their mark is in the range 50 to 64, and *D* if their mark is below 50.

Write a program that repeatedly reads marks until the user presses the “cancel” button, and then prints out the number of students who have got *As*, *Bs*, *Cs* and *Ds*. For example, given the following marks, 70, 80, 90, 50, 30, 70, 90, 85, 45, 67, the program should print out:

The number of *As*: 4The number of *Bs*: 3The number of *Cs*: 1The number of *Ds*: 2

```
import javax.swing.*;

public class ExamResults {

    public static void main(String args[]) {
```

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) [10 marks] Counting spaces

Write a program to read a line of text, using an input dialog box, and print the number of spaces in the input. For example, if the input is “This is a line of text!” (where □ indicates a space), the program should print out:

The number of spaces: 5

```
import javax.swing.*;

public class CountSpaces {

    public static void main(String args[]) {
```

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 4. Programs with objects

[20 marks]

Consider the following program:

```
public class TestCodeBank {
    public static void main (String[] args) {
        Bank b = new Bank("john", 01, 1000);
        System.out.println(b.getBalance());
        b.deposit(30);
        System.out.println(b.getBalance());

        Bank account = new Bank("peter", 05, 300);
        System.out.println(account.getBalance());
        account.deposit(4000);
        System.out.println(account.getBalance());
    }
}

class Bank {
    private double balance;
    private String name;
    private int accountID;

    public Bank(String n, int ID, double d){
        name = n;
        accountID = ID;
        balance = d;
    }

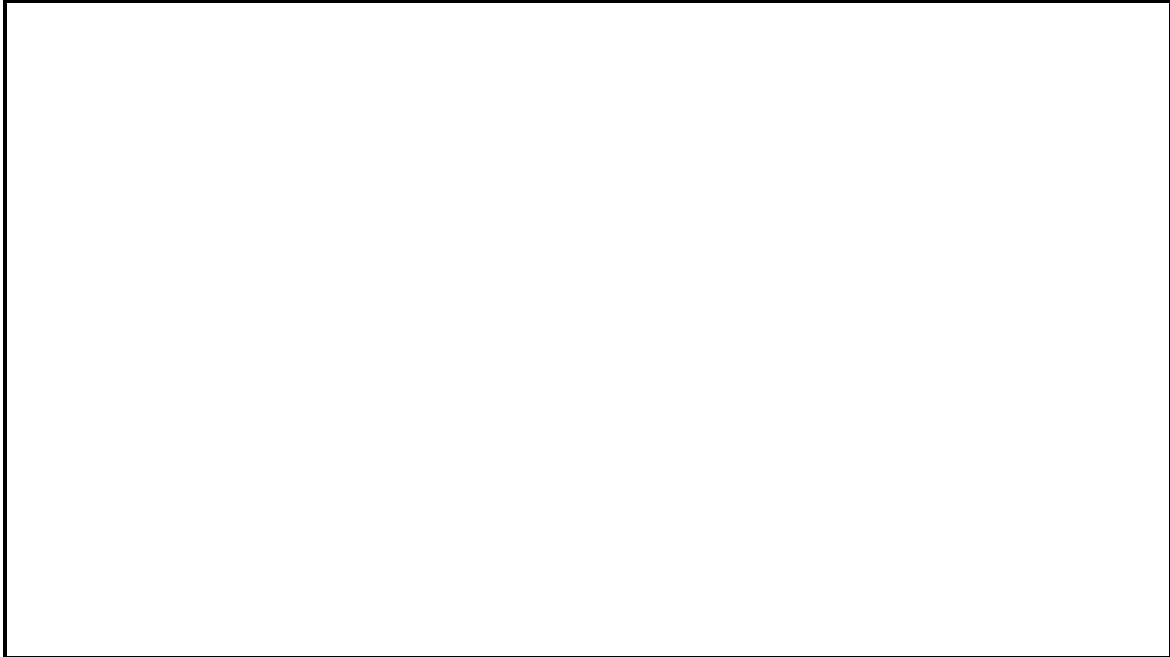
    public double getBalance(){
        return balance;
    }

    public void deposit(double x){
        balance = balance + x;
    }
}
```

(a) [8 marks] What output will this program produce?

(b) [12 marks]

Add a `withdraw` method to the `Bank` class. This method has one parameter which gives the amount of money to withdraw. If there is enough money in the account, it should withdraw money and print the new balance, otherwise, it should print ```Sorry, you have not got enough money```.



Question 5. Programs with arrays

[25 marks]

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

```
private int[ ] nums = new int[5];
```

For question (a) and (b), suppose the array holds 45, 16, 25, 80, 50 and the array looks like this:

<code>nums[0]</code>	<code>nums[1]</code>	<code>nums[2]</code>	<code>nums[3]</code>	<code>nums[4]</code>
45	16	25	80	50

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

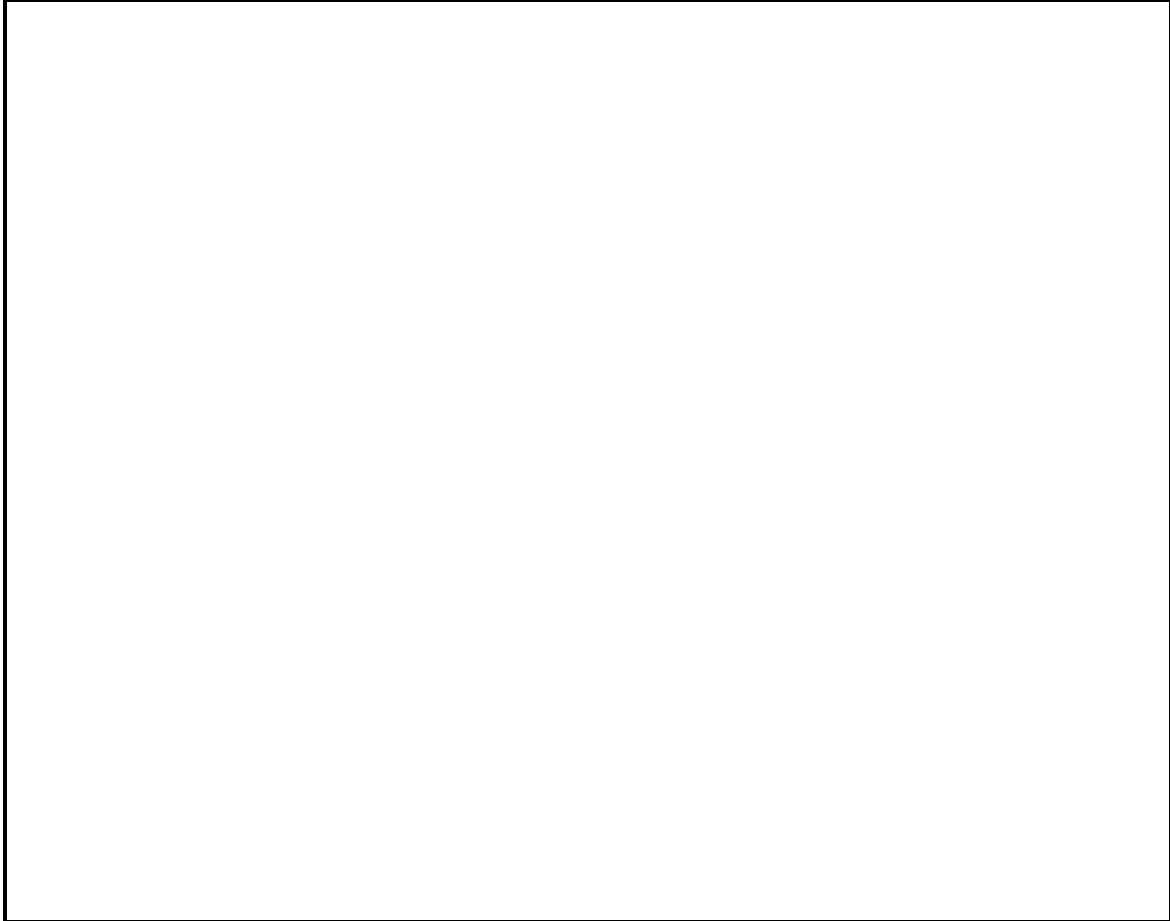
```
public void print1() {
    int i = 1;
    System.out.println(nums[2 * i + 1]);
    System.out.println(2 * nums[i] + 1);

    for (int k = 0; k <= nums.length/2 ; k++)
        System.out.println(nums[k]);
}
```

(b) [6 marks] What will the following method print out?

```
public void print2() {
    int c = 0;
    for (int i = 0; i < nums.length - 1 ; i++) {
        if (nums[i] > nums[i+1]) {
            System.out.println(nums[i]);
            c++;
        }
    }
    System.out.println(c);
}
```

(c) [14 marks] Write a method `findDuplicates` (to be added to the class) to detect whether there are duplicated numbers in the array. It should return `true` if (at least) two numbers are the same, and return `false` if the numbers are all different. If the array holds 45, 16, 25, 80, 50, it should return `false`. If the array holds 45, 16, 45, 80, 45 or 45, 16, 25, 16, 50, it should return `true`.



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.