TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

**EXAMINATIONS — 2005**

END-OF-YEAR

**COMP 102**

**INTRODUCTION TO COMPUTER PROGRAM DESIGN**

**Time Allowed:** 3 Hours

**Instructions:** Attempt ALL Questions.

Answer in the appropriate boxes if possible — if you write your answer elsewhere, make it clear where your answer can be found.

The exam will be marked out of 180 marks.

Non-programmable calculators without a full alphabetic key pad are permitted.

Non-electronic foreign language dictionaries are permitted.

There are spare pages for your working and your answers in this exam.

## Questions

| | **Marks** |
|---|---|
| 1. Understanding Java programs | [45] |
| 2. Writing simple Java Programs | [25] |
| 3. Classes and objects | [20] |
| 4. Arrays of objects | [15] |
| 5. Inheritance | [15] |
| 6. Files | [20] |
| 7. GUI | [20] |
| 8. Recursion | [20] |

Student ID: .....

## Question 1. Understanding Java programs [45 marks]

In parts **(a)** to **(g)**, you should assume each method is declared in a suitable class and called on an object of this class. The name and other details of this class are unimportant as the methods do not refer to fields or other methods of the class.

**(a)** [3 marks] What will be printed when the following method `Q1a` is called?

```
public void Q1a() {
  int x = 4;
  int y = 3;
  int z = x*y;
  x = x+y;
  y = x-y;
  System.out.println("x = " + x);
  System.out.println("y = " + y);
  System.out.println("z = " + z);
}
```

**(b)** [4 marks] What will be printed when the following method `Q1b` is called?

```
public void Q1b() {
  String s = "javax.swing.*";
  System.out.println( s.length() );
  System.out.println( s.charAt(7) );
  int k = s.indexOf('a');
  System.out.println( k );
  System.out.println( s.indexOf('a', k+1) );
}
```

**(c)** [6 marks] Consider the following method definition:

```
public void Q1c(int a, int b) {
  int c = 0;
  int d;
  if ( a < b) {
    c = a+b;
  }
  d = 2*b;
  if ( a > c )
    c = a;
  else
    d = b;
  System.out.println(c + " " + d);
}
```

**(i)** [2 marks] What will be printed when method `Q1c` is called with **5** and **0** as its arguments?

**(ii)** [2 marks] What will be printed when method `Q1c` is called with **3** and **3** as its arguments?

**(iii)** [2 marks] What will be printed when method `Q1c` is called with **2** and **1** as its arguments?

Student ID: .....

**(d)** [5 marks] What will be printed when the following method `Q1d` is called?

```
public void Q1d() {
  int x = 0;
  int y = 30;
  while ( x < y ) {
    x = x + 2;
    y = y - x;
    System.out.println( x + " " + y);
  }
}
```

**(e)** [4 marks] What will be printed when the following method `Q1e` is called?

```
public void Q1e() {
  char[] a = {'a', 'b', 'c', 'd', 'e', 'f'};
  for (int k = 0; k < a.length/2; k++) {
    System.out.println(a[k] + " " + a[2*k+1]);
  }
}
```

Student ID: .....

**(f)** [8 marks] What will be printed when the following method `Q1f` is called?

```
public void Q1f() {
  int[] a = new int[8];
  for (int k = 0; k < a.length; k++) {
    a[k] = 8-k;
    System.out.print(a[k] + " ");
  }
  System.out.println();
  int s = 0;
  for (int k = 0; k < a.length; k=k+2) {
    s = s + a[k];
    System.out.println(s);
  }
}
```

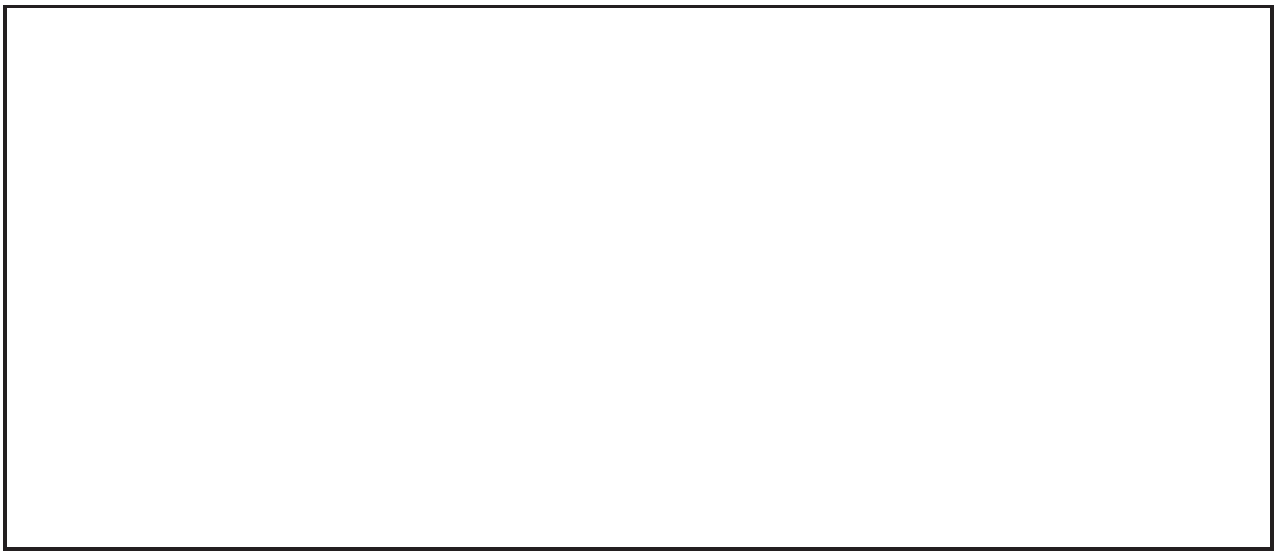
**(g)** [9 marks] What will be printed when the following method `Q1g` is called?

```
public void Q1g() {
  int m = 3;
  int[][] a = new int[m][m];
  for (int i = 0; i < m; i++) {
    for (int j = 0; j < m; j++) {
      a[i][j] = m*i-j;
      System.out.print(a[i][j] + " ");
    }
    System.out.println();
  }
  for (int i = 0; i < m; i++) {
    System.out.print(a[i][m-i-1] + " ");
  }
  System.out.println();
}
```

Student ID: .....

**(h)** [6 marks] What will the following program print out?

```
public class Q1h {
  public static void main(String[] args) {
    Thing[] things  = new Thing[3];
    things[0] = new Thing(0, 2);
    things[1] = new Thing(1, 3);
    things[2] = new Thing(4, 5);
    for (int k = 0; k < things.length; k++) {
      System.out.println(things[k].f() + " " + things[k].g());
    }
  }
}

class Thing {
  private int x, y;
  public Thing(int xx, int yy) {
    x = xx;
    y = yy;
  }
  public int f() {
    return x+y;
  }
  public int g() {
    return x*y;
  }
}
```



Student ID: .....

**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.

Student ID: .....

## Question 2. Writing simple Java programs [25 marks]

**(a)** [15 marks] **Strings and conditionals**

You are to write a program that reads a string and analyses it, as follows:

- If the string is empty, the program should print `NULL`.
- If the string starts and ends with a digit, the program should print `NUMBER`.
- If the string starts and ends with a letter, the program should print `WORD`.
- Otherwise, the program should print `OTHER`.

**Complete the following program so that it behaves as described above.**

You should print output using `System.out.println`.

You can test whether a character `c` is a digit with the call `Character.isDigit(c)`, and you can test whether `c` is a letter with the call `Character.isLetter(c)`.

```
import javax.swing.*;
public class StringAnalyser {
  public static void main(String[] args) {
    // Read and analyse a string
    String s = JOptionPane.showInputDialog("Enter string");









  }
}
```

Student ID: .....

**(b)** [10 marks] **Loops and arrays**

You are to write a method, `checkArray`, that takes a two dimensional array of integers as its argument and prints a message showing the largest value in the array and the number of zeros in the array.

For example, if the method was called as follows:

```
int[][] vals = { { 1, 0, 2, 9 },
                 { 0, -1, 9, 6 },
                 { 3, 0, 0, 8 } };
checkArray(vals);
```

the output would be something like:

```
Largest value = 9
Number of zeros = 4
```

**Complete the following method so that it behaves as described above.**

You should print output using `System.out` methods.

```
public void checkArray(int[][] a) {









}
```

Student ID: .....

## Question 3. Classes and Objects [20 marks]

You are required to complete the `Student` class shown on the facing page.

The `Student` class is intended to be used for recording and printing student marks. The `Student` constructor takes as arguments the name of the student and the number of marks to be recorded. The `enterMark` method takes one mark as its argument and records it. The `printResults` method checks whether the correct number of marks have been recorded — if so, `printResults` prints the student's name, marks and total mark; if not, it prints an error message showing the student's name and the number of marks that were recorded.

For example, executing the program shown below on the left should produce the output shown below on the right.

```
public class StudentMarks {
  public static void main(String[] args) {
    Student s = new Student("Bill", 3);
    s.enterMark(3);
    s.enterMark(5);
    s.enterMark(2);
    s.printResults();
    Student t = new Student("Fred", 3);
    t.enterMark(4);
    t.enterMark(1);
    t.enterMark(7);
    t.printResults();
    Student x = new Student("Chris", 3);
    Student y = new Student("Dave", 3);
    y.enterMark(0);
    x.enterMark(0);
    y.enterMark(1);
    x.enterMark(1);
    y.enterMark(2);
    y.enterMark(3);
    x.printResults();
    y.printResults();
  }
}
```

```
Bill
  Marks: 3 5 2
  Total: 10
Fred
  Marks: 4 1 7
  Total: 12
Error: Chris has 2 marks
Error: Dave has 4 marks
```

You should print output using `System.out` methods.

You should think carefully about what fields the `Student` class needs in order to record information about a student, and how these fields should be used and/or updated by the constructor and the two methods.

Student ID: .....

```
public class Student {









    public Student(String s, int n) {









    }

    public void enterMark(int m) {









    }

    public void printResults() {

















    }
}
```

Student ID: .....

## Question 4. Arrays of objects [15 marks]

You are required to write a program that reads the name and price of several products, and then prints the name and price of each product.

The required program consists of two classes: `ProductProgram` and `Product`. The `Product` class (shown below) has already been written — it is designed to read, store, print and access information about products. You are also given an outline for `ProductProgram` (shown on the facing page), which you are to complete.

The program is intended to read the number of products in the class, create an array of `Product`s of that size, then read the name and price of each product and store them in the array. It should then go through the array and print the name and price of each product.

Complete the definition of `ProductProgram` so that it behaves as described above.

**Your program must use the `Product` class to read, store, print and access information about products.**

You must not change the `Product` class.

```
class Product {

    private String name;
    private double price;

    public Product() {
        name = JOptionPane.showInputDialog("Enter product name");
        String s = JOptionPane.showInputDialog("Enter price for " + name);
        price = Double.parseDouble(s);
    }

    public void print() {
        System.out.println(name + "   " + price);
    }

    public void setName(String s) {
        name = s;
    }

    public String getName() {
        return name;
    }

    public void setPrice(double p) {
        price = p;
    }

    public double getPrice() {
        return price;
    }
}
```

Student ID: .....

```
import javax.swing.*;
public class ProductProgram {
  public static void main(String[] args) {
    // Read n
    String s = JOptionPane.showInputDialog("How many products?");
    int n = Integer.parseInt(s);
    // Create an array of Products, of size n









    // Read product data and store it

















    // Print data for each product

  }
}
```

Student ID: .....

**Question 5. Inheritance** [15 marks]

Consider the `Product` class in the previous question. You are required to write a new class `SaleProduct` in this question using inheritance.

- The `SaleProduct` class has an extra data field `salePrice`.
- When you create a `SaleProduct` object, the constructor first asks for the name and price of the product which is exactly the same as in creating a `Product` object, then it asks for the sale price for this product, and finally it updates the price of this product to be the same as the sale price.
- When you print the details of a `SaleProduct`, as well as printing the name and the price, it also prints a line saying "`Special!`".

You must use inheritance and you must **NOT** change the `Product` class.

Student ID: .....

**(a)** [10 marks] Write your `SaleProduct` class here.

**(b)** [5 marks] Suppose we want to use the `ProductProgram` class in the previous question to store and print details of sale products instead of products. You only need to change one line in the `ProductProgram` class. Write down the new line.

Student ID: .....

**Question 6. Files** [20 marks]

Consider the following program:

```
import java.io.*;
import java.util.*;

public class Files {
  public static void main(String[] args) {

    try {
      Scanner inFile = new Scanner(new File("data.txt"));

      FileWriter outStream = new FileWriter("out.txt");
      PrintWriter outFile = new PrintWriter(outStream);

      while(inFile.hasNext()) {
        double marks = inFile.nextDouble();
        System.out.println(marks);
        outFile.println(marks);
      }
      inFile.close();
      outFile.close();
    }
    catch(InputMismatchException ex) {
      System.out.println("wrong data format");
    }
    catch(IOException ex) {
      System.out.println("File I/O error");
    }
  }
}
```

Suppose the program is run with the following file `data.txt`:

```
40.0 50.0
45.5
67.0 wow!
46.6 30.3 70.7 test this again
30.5 40.6
20.5 60.7 nearly there 40.6 20.9
65.0 60.2 good
```

Student ID: .....

**(a)** [5 marks] What will the program `Files` print out to the screen?

**(b)** [5 marks] What data is saved to the output file `out.txt`? If no data is saved, explain why.

Student ID: .....

**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.

Student ID: .....

**(c)** [10 marks] Modify this program so it writes all the numbers into the new file, one number on each line. Make sure your program ignores all the noise (words, commments, etc).

```
import java.io.*;
import java.util.*;

public class Files {
  public static void main(String[] args) {

    try {
      Scanner inFile = new Scanner(new File("data.txt"));

      FileWriter outStream = new FileWriter("out.txt");
      PrintWriter outFile = new PrintWriter(outStream);









      while(inFile.hasNext()) {


        double marks = inFile.nextDouble();


        System.out.println(marks);


        outFile.println(marks);


      }









      inFile.close();
      outFile.close();
    }
    catch(NumberFormatException ex) {
      System.out.println("wrong number format");
    }
    catch(IOException ex) {
      System.out.println("File I/O error");
    }
  }
}
```

Student ID: .....

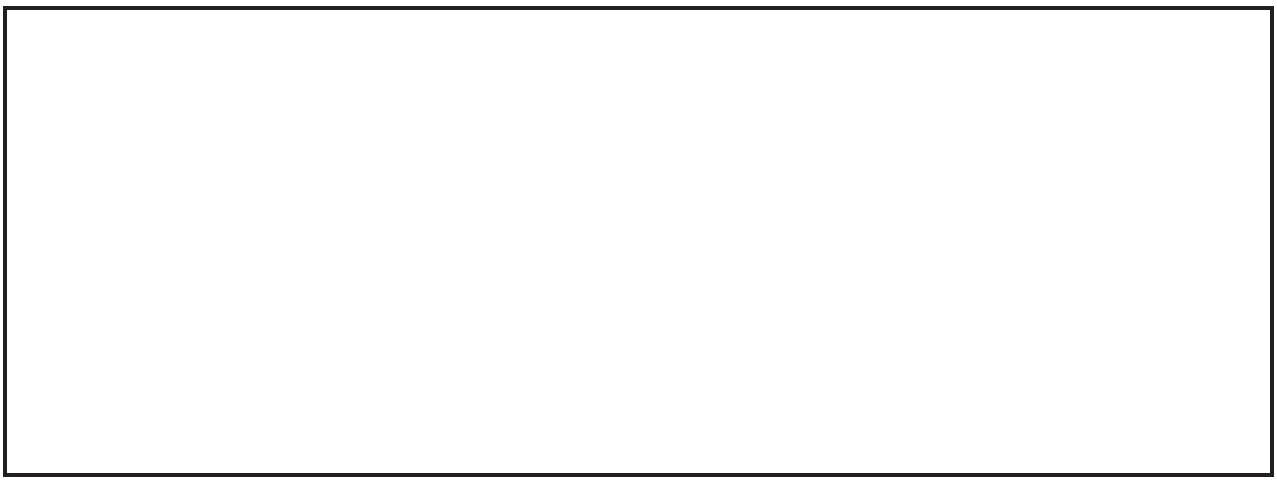
**Question 7. GUI** [20 marks]

Consider the following program:

```
import javax.swing.*;
import java.awt.event.*;
public class Calculator implements ActionListener {
  private JFrame frame;
  private JPanel panel;
  private JTextField screen;
  private JButton mul;
  private JButton add;
  private JButton eq;
  private int lhs = 0;
  private int rhs = 0;
  private Object prevButton;

  public Calculator() {

    frame = new JFrame("Simple Calculator");
    panel = new JPanel();
    screen = new JTextField(8);
    mul = new JButton("*");
    add = new JButton("+");
    eq = new JButton("=");

    panel.add(screen);
    panel.add(mul);
    panel.add(add);
    panel.add(eq);
    frame.getContentPane().add(panel);

    mul.addActionListener(this);
    add.addActionListener(this);
    eq.addActionListener(this);

    frame.setSize(280,70);
    frame.setVisible(true);
  }
  public void actionPerformed(ActionEvent e) {
    String input = screen.getText();
    int rhs;
    Object button = e.getSource();
    screen.setText("");
    if (button == eq) {
      rhs = Integer.parseInt(input);
      if(prevButton == add)
        screen.setText(Integer.toString(lhs + rhs));
      else
        screen.setText(Integer.toString(lhs * rhs));
    } else {
      lhs = Integer.parseInt(input);
      prevButton = button;
    }
  }
  public static void main(String args[]) {
      new Calculator();
  }
}
```

Student ID: .....

**(a)** [8 marks] Draw a picture to show the GUI this program creates. You may assume the window is big enough to hold all the components.



**(b)** [12 marks] The following shows a number of example cases for testing this program. For example,

[ 10 ] [ + ] [ 5 ] [ = ]

means that the user first types `10` in the text field, then presses the [ + ] button, then types number `5` in the text field, and finally presses the [ = ] button. Your task is to give the output in the text field. The first one is done for you.

| | |
|---|---|
| [ 10 ] [ + ] [ 5 ] [ = ] | [ 15 ] |
| [ 10 ] [ + ] [ 5 ] [ = ] [ * ] [ 2 ] [ = ] | [ ] Your answer here |
| [ 10 ] [ + ] [ 5 ] [ * ] [ 2 ] [ = ] | [ ] Your answer here |
| [ 10 ] [ + ] [ 5 ] [ * ] [ 2 ] [ + ] [ 7 ] [ = ] | [ ] Your answer here |
| [ 10 ] [ + ] [ 5 ] [ = ] [ * ] [ 2 ] [ = ] [ + ] [ 7 ] [ = ] | [ ] Your answer here |

Student ID: .....

## Question 8. Recursion [20 marks]

Here is the recursive method that solves the Tower of Hanoi problem:

```
public void tower(char from, char to, char help, int n) {

  System.out.println("n :" + n);

  if (n==1)
    System.out.println("--- Move disk 1 from " + from + " to " + to);

  else {
    tower(from, help, to, n-1);

    System.out.println("*** Move disk " + n + " from " + from +" to " + to);

    tower(help, to, from, n-1);
  }
}
```

Note: The method prints the argument `n` each time the method is called, and it prints the instructions.

What will the method print out if it is called as follows:

**(a)** [3 marks]

```
tower('A', 'C', 'B', 1)
```

Student ID: .....

**(b)** [7 marks]

```
tower('A', 'C', 'B', 2)
```

**(c)** [10 marks]

```
tower('A', 'C', 'B', 3)
```

**********

Student ID: .....

**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.

Student ID: .....

**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.