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

Student ID: ....

EXAMINATIONS — 2010

END-OF-YEAR

COMP 102
INTRODUCTION TO COMPUTER PROGRAM DESIGN

Time Allowed: 3 Hours ******** WITH SOLUTIONS **********

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.

Java Documentation will be provided with the exam script

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

Questions

	Marks
1. Understanding Java	[40]
2. Event driven input	[15]
3. Files	[25]
4. Arrays	[45]
5. Arrays of Objects	[40]
6. Interface classes	[15]

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 1. Understanding Java [40 marks]

(a) [5 marks] What will the following fragment of Java print out?

```
int x = 2;
int y = 4;
y = y + 1;
x = y;
System.out.println("x=" + x + "  y=" + y + "  " + (x * y));
y = x;
y = y + 1;
System.out.println("x=" + x + "  y=" + y);
```

```
x=5 y=5 25
x=5 y=6
```

(b) [6 marks] Consider the following choice method (note the **if**'s and **else**'s carefully):

```
public void choice(int x, int y) {
    if (x < 10 && y > 20){
        System.out.print("1st ");
    }
    if (x >= 5 || y > 10) {
        System.out.print("2nd ");
    }
    else if (x < 10 && y==5){
        System.out.print("3rd ");
    }
    else {
        System.out.print("4th ");
    }
    System.out.println ();
}
```

What would the following calls to choice print out?

choice(3, 5) ⟹ 3rd

choice(5, 25) ⟹ 1st 2nd

choice(4, 6) ⟹ 4th

(Question 1 continued on next page)

(Question 1 continued)

(c) [5 marks] What will the following fragment of Java print out?

```
        int i = 2;
        int k = 1;
        while (k <= 10){
            System.out.println(k + "  " + (i * k));
            k = k + 3;
        }
    System.out.println("Done");
```

```
1 2
4 8
7 14
10 20
Done
```

(d) [8 marks] Consider the following two methods. If the testNumbers method is called, what will it print out?

```
public int twoNumbers(int x, int y){
    if (x > y)
        return x;
    else
        return y + 10;
}
public void testNumbers(){
    System.out.println(this.twoNumbers(10, 5));
    int x = 2;
    int y = this.twoNumbers(x,10) + 1;
    System.out.println(this.twoNumbers(x, y));
    System.out.println(this.twoNumbers(y, x));
}
```

```
10

31

21
```

(e) [8 marks] What will the following fragment of Java print out if the user enters the following in response to the prompt:

```
It has 15 circles and 2 triangles but no rings
```

```
Scanner scan = new Scanner(System.in);
System.out.println("Input: ");
while ( true ){
    String word = scan.next();
    if ( word.equals("but"))
        break;
    if (scan.hasNextInt()) {
       System.out.println(scan.nextInt ());
    }
    else {
        System.out.println(word);
    }
}
System.out.println("Done");
```

```
Input:  It has 15 circles and 2 triangles but no rings
It
15
circles
2
triangles
Done
```

(f) [8 marks] Complete the following definition of a Shoe class. Shoe objects should have two fields called brand and size. brand should hold a String, and size should hold an integer to specify the size of the shoe.

The class should have a constructor that takes two arguments and sets the fields to its arguments. The class should have one method called toString (with no parameters) to return a String that contains the values of both fields.

```
public    class Shoe {
        private String brand;
        private int size;

        public Shoe(String b, int s){
             this.brand = b;
             this.size = s;
        }

        public String toString(){
             return this.brand + "  " + this.size + "  ";
        }
}
```

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.

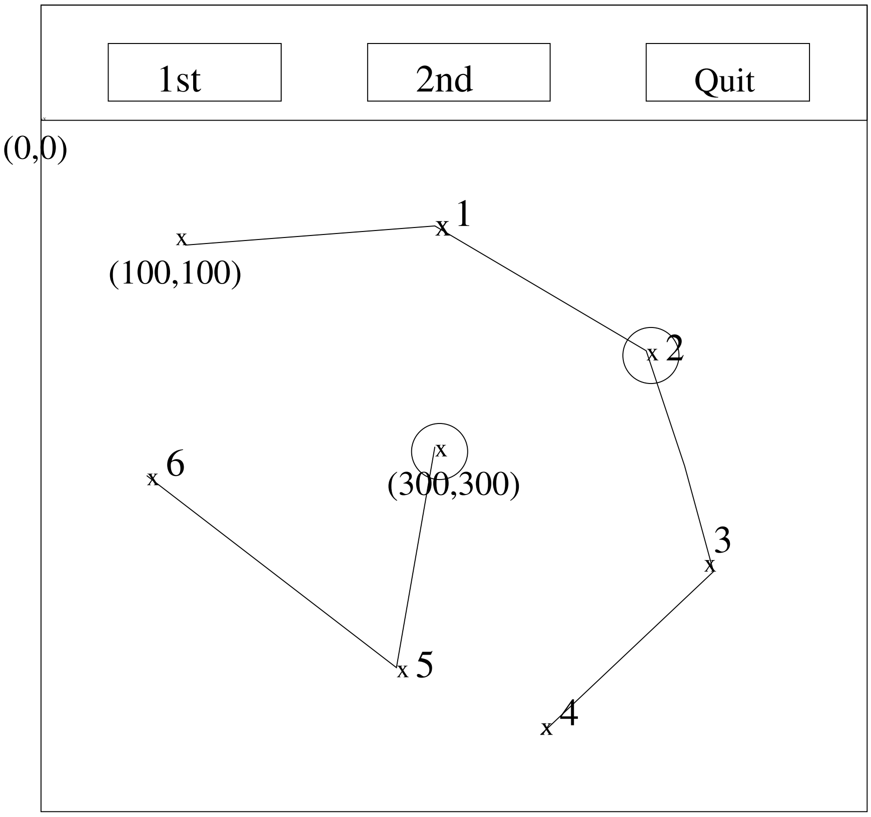
Question 2. Event-Driven Input [15 marks]

Consider the PatternWriter class on the facing page, which constructs a simple GUI with three buttons and a canvas that responds to the mouse. It has two fields that store a position (xPos, yPos), and the initial position is at (100, 100).

Sketch below what the program would draw on the canvas if the user took the following actions in sequence:

1. press and release mouse at point 1
2. press and release mouse at point 2
3. press the "1st" button
4. press and release mouse at point 3
5. press and release mouse at point 4
6. press the "2nd" button
7. press and release mouse at point 5
8. press and release mouse at point 6

please note that the two positions with coordinates (100,100) (300,300) are given in the figure.



(Question 2 continued on next page)

(Question 2 continued)

```
public class PatternWriter implements ActionListener, MouseListener{
    private JFrame frame;
    private DrawingCanvas canvas;
    private int xPos = 100;
    private int yPos = 100;

    public PatternWriter(){
        this.frame = new JFrame("TestGui");
        this.frame.setSize(600, 600);
        this.canvas = new DrawingCanvas();
        this.canvas.addMouseListener(this);
        this.frame.getContentPane().add(this.canvas, BorderLayout.CENTER);

        JPanel panel = new JPanel();
        this.frame.getContentPane().add(panel, BorderLayout.NORTH);
        JButton button = new JButton("1st");
        button.addActionListener(this);
        panel.add(button);
        JButton cButton = new JButton("2nd");
        cButton.addActionListener(this);
        panel.add(cButton);
        JButton qButton = new JButton("Quit");
        qButton.addActionListener(this);
        panel.add(qButton);
        this.frame.setVisible(true);
    }
    public void actionPerformed(ActionEvent e){
        String cmd = e.getActionCommand();
        if (cmd.equals("1st") ){
            this.canvas.drawOval(this.xPos−10, this.yPos−10, 20, 20);
        }
        else if (cmd.equals("2nd") ){
            this.xPos = 300;
            this.yPos = 300;
            this.canvas.drawOval(this.xPos−10, this.yPos−10, 20, 20);
        }
        else if (cmd.equals("Quit") ){
            this.frame.dispose();
        }
    }
    public void mousePressed(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {
        this.canvas.drawLine(this.xPos, this.yPos, e.getX(), e.getY ());
        this.xPos = e.getX();
        this.yPos = e.getY();
    }
    public void mouseClicked(MouseEvent e) {}
    public void mouseEntered(MouseEvent e) {}
    public void mouseExited(MouseEvent e) {}
}
```

Question 3. Files [25 marks]

Suppose the file `Data.txt` contains data about tutor working hours. Each line of the file contains information about one tutor: the tutor's name, the hours worked during a week (up to 5 integers or none). For example, the following might be the first five lines of the file.

```
Alan 5 6
Bob 3 1 4 2 1
Frank
Kris 2 3
Peter 4
```

(a) [7 marks] What would the following method print out if `Data.txt` only had the five lines shown above?

```
public void printFile (){
    try{
        Scanner scan = new Scanner (new File("Data.txt"));

        while ( scan.hasNext() ){
            System.out.println (scan.next ());
            if (scan.hasNextInt()){
                scan.nextInt ();
            }
        }
        scan.close ();
    }
    catch(IOException e){System.out.println("File reading failed");}
}
```

```
Alan
6
Bob
1
2
Frank
Kris
3
Peter
```

(b) [8 marks] Complete the following printNames method to print out all the tutor names in the file and ignore other information. For example, if the file only had the five lines in the example above, printNames should print

```
Alan
Bob
Frank
Kris
Peter
```

```
public void printNames(){
    try{
        Scanner scan = new Scanner (new File("Data.txt"));
         while ( scan.hasNext() ){
             System.out.println(scan.next ());
             if (scan.hasNextInt()){
                 scan.nextLine();
             }
         }








        scan.close();
    }
    catch(IOException e){System.out.println("File reading failed");}
}
```

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.

(c) [10 marks] Complete the following printSummary method to print out the name and the total working hours for each tutor. It should also print out the name and working hours of the tutor who has worked the most of hours. If there is a tie, you may just print one of them.

For example, if the file only had the five lines in the example above, pritnSummary() should print out

```
Alan:11
Bob:11
Frank:0
Kris:5
Peter:4
Tutor with most hours:  Alan 11
```

```
public void printSummary(){
    try{
        Scanner scan = new Scanner (new File("Data.txt"));
        int max = 0;
         String name="";
         while ( scan.hasNext() ){

            String s = scan.next();
            int h = 0;
            while (scan.hasNextInt()){
                h = h + scan.nextInt ();
            }
            System.out.println(s + ":" + h);
            if (h > max) {
                name =s;
                max = h;
            }
        }
        System.out.println("Tutor with most hours " + name + "  :  " + max);




        scan.close();
    }catch(IOException e){System.out.println("fail");}
}
```

Question 4. Array

[45 marks]

(a) [9 marks] For each of the following statements, write code to declare and create an array to represent the data. If you need any surpporting classes, state the name of the class, and specify the types and the names of its data fields. The first one is done for you as an example. You may make your own assumptions about array size if necessary.

(i) students' test marks for a class of 40 students

```
double[] marks = new double[40]

no supporting class is needed
```

(ii) students' names for a class of 25 students.

```
String[] names = new String[25]
no supporting class is needed
```

(iii) credit-card transactions that contain a transaction number, a merchant name and a charge.

```
Transaction[] card = new Transaction[100]
class Transaction
private int No;
private String name;
private double charge;
```

(iv) students' names for a class and homework grades for each student.

```
2D or array of Objects
String[][] table = new String[50][10];
or Studnet[] list = new Student[50];
class Student
private String name;
private String[] grades;
```

(b) [8 marks] Suppose the variable words is declared and intialised as follows:

```
String [ ] words = new String [ ] {"dog", "bee", "fox", "cat", "ant", "eel"};
```

words:	dog	bee	fox	cat	ant	eel
	0	1	2	3	4	5

What will the following code fragment print out?

```
System.out.println(words[3]);
System.out.println(words.length);
System.out.println(words[4].length ());
for( int j = words.length−1; j >= 0; j−−){
    System.out.print(words[j]+ "  ");
}
System.out.println ();
```

```
cat
6
3
eel  ant  cat  fox bee  dog
```

(c) [10 marks] What will the following method print out?

```
public void test2D(){
    int [][] nums = new int [][] {{2, 3, 5, 1},
                                  {4, 2, 5, 7},
                                  {8, 1, 0, 2}};
   System.out.println(nums[1][3]);
   System.out.println(nums.length);
   System.out.println(nums[0].length);

   for( int r = 0; r < nums.length; r++) {
       int total =0;
       for( int c = 0; c < nums[r].length; c++) {
           System.out.print(nums[r][c]+ "  ");
           total = total + nums[r][c];
       }
       System.out.println( total );
  }
}
```

```
7
3
4
2  3  5  1  11
4  2  5  7  18
8  1  0  2  11
```

(d) [8 marks] Complete the following mean method. mean has one parameter – an array of int – and should return the mean of the values in the array. You may assume that the array is not null and the length of the array is not zero.

```
public double mean(int [ ] data){

        double total = 0;
        for ( int i=0; i<data.length; i++){     //OR   for (int n : data){
              total = total + data[i];          //          total += n;
        }
        return total /data.length;

}
```

(e) [10 marks] Complete the following reverse method. reverse has one parameter – an array of int – and should reverse the order of the values in the array. For example, given the array {10, 4, 3, 7, 8, 5} it should change it to be {5, 8, 7, 3, 4, 10}. You may assume that the array is not null.

```
public void reverse(int [ ] data){

        for( int i=0; i<data.length/2; i++) {
              int temp = data[i];
              data[i] = data[data.length−1−i];
              data[data.length−1−i] = temp;
        }

}
```

Question 5. Arrays of Objects [40 marks]

This question concerns an AccountManager class to manage a list of accounts. The class stores the information about the accounts in a field containing an array of Account objects. It also has a count field that contains the number of Accounts, and the Accounts are stored in cells 0 through count-1 of the array. The AccountManager class includes methods to list all the accounts, add an account to the list, and remove a specified account, etc.

Part of the Account class, for representing information about individual accounts, is shown below.

```
public class Account{
    private int no;
    private double balance;

    public Account(int n, double d){
        this.no = n;
        this.balance = d;
    }

    public String toString(){
        return this.no + "  " + this.balance;
    }

    public int getNo(){
        return this.no;
    }

    public double getBalance(){
        return this.balance;
    }

    public void deposit(double m){
        this.balance = this.balance + m;
    }

    public void withdraw(double m) {
        this.balance = this.balance −m;
    }
}
```

(Question 5 continued)

The following are some of the fields and one of the methods of the AccountManager class:

```
public class AccountManager{

    private int maxAccounts = 30;
    private int count = 0;
    private Account[] accountList = new Account[maxAccounts];

    public void testAccounts(){
        Account a1 = new Account(123, 1000.0);
        Account a2 = new Account(146, 3000.0);
        Account a3 = new Account(244, 2000.0);

        this.accountList[0] = a1;
        this.count++;
        this.accountList[this.count] = a2;
        this.count++;

        System.out.println("1:  " + a3.getBalance());
        System.out.println("2:  " + this.accountList[0].toString ());
        System.out.println("3:  " + this.accountList[1].getNo());

        this.accountList[1].deposit(500.0);

        for( int i=0; i<this.count; i++){
            System.out.println(i + ":  " + this.accountList[i]. toString ());
        }
    }
}
```

(a) [8 marks] What will the testAccounts method print out?

```
1:  2000.0
2:  123 1000.0
3:  146
0:  123 1000.0
1:  146 3500.0
```

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] Complete the following displayAccounts method of the AccountManager class which should print the details of each account, including the account number and the account balance.

```
public void displayAccounts(){


    for ( int i=0; i<this.count; i++){
        System.out.println(this.accountList[i].toString());
    }







}
```

(c) [10 marks] Complete the following findAccountIndex method of the AccountManager class which should return the index of an Account object with the specified account number. If the account is not found, it should return -1.

```
public int findAccountIndex(int n){


    for ( int i=0; i<this.count; i++){
        if (this.accountList[i].getNo()==n){
            return i;
        }
    }
    return −1;





}
```

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.

(d) [12 marks] Complete the following deleteAccount method in the AccountManager class which should remove the account of the specified accoung number.

It is important that there should be no nulls in the array in the range from 0 to count-1. The order the accounts stored in the list is important and should not be changed.

The deleteAccount method should print a message if the account is not found.

Hint: You may use the method defined in the previous question if you wish.

```
public void deleteAccount(int n){

    int k = this.findAccount(n);
    if (k>-1) {
        for( int i = k; i < this.count-1; i++){
            this.accountList[i] = this.accountList[i+1];
        }
        this.count--;
    }
    else {
        System.out.println("not found");
    }


}
```

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 6. Designing with Interfaces [15 marks]

Assignment 10 has many classes such as Line and Dot to implement the Shape interface. Suppose your task is to add more classes such as Triangle, Ring and Star to this program in the same way so your program can handle more shapes.

Here is the Shape inteface given in the assignment.

```
public interface Shape{

  public boolean pointOnShape(int x, int y);
  public void moveBy(int dx, int dy);
  public void render(DrawingCanvas canvas);
  public void resize( int changeWd, int changeHt);
  public String toString ();
}
```

(a) [7 marks] Suppose you have added three new classes Triangle, Ring and Star to this program using the Shape interface. Comparing the three new classes mentioned, state their similarities. (You don't need to give any code)

They all implement the shape interface
They all have the five methods defined in the interface and
the headings are the same.
The body of these methods are different.
They can have other methods of their own.

(b) [8 marks] Explain the main advantages of using a Shape interface in this program.

Using an interface, you can create an array of the type
Shape to save
all the different shapes.
It is easier to add a new shape. You only need to add one
more button
and change the actionPerformed method to reponse to this
button.
Anything else stay the same.

****