

THE UNIVERSITY OF AUCKLAND

SUMMER SEMESTER, 2008
Campus: City

COMPUTER SCIENCE

Principles of Programming

(Time allowed: TWO hours)

NOTE: Attempt **ALL** questions
Write your answers in the space provided
There is space at the back for answers that overflow the allotted space
No calculators are permitted

| | |
|---------------------------|--|
| Surname: | |
| Forenames: | |
| Student ID number: | |
| Login name: | |

| | | | |
|--------------------|--------------------|--------------------|----------------------------|
| Q1 (/20) | Q4 (/6) | Q7 (/6) | Q10 (/8) |
| Q2 (/8) | Q5 (/6) | Q8 (/8) | Q11 (/10) |
| Q3 (/8) | Q6 (/10) | Q9 (/10) | TOTAL (/100) |

CONTINUED

Question/Answer Sheet

ID:

Question 1 (20 marks)

a) What is the output of the following code?

```
System.out.println(2 + 9 + "3" + 5 + (2 + 4));
```

11356

(2 marks)

b) What is the output of the following code?

```
System.out.println(Math.max(3,5) + Math.min(4,9));
```

9

(2 marks)

c) Write a Java boolean expression which tests whether the String identified by the variable, name, is less than 10 characters long and contains the letter 'r'.

name.length() < 10 && name.indexOf('r') > -1

(2 marks)

Question/Answer Sheet

ID:

d) Complete the output of the following code.

```
Rectangle a = new Rectangle(0, 1, 2, 3);
Rectangle b = new Rectangle(10, 20, 30, 40);
Rectangle c;

c = b;
b = new Rectangle(10, 20, 30, 40);

System.out.println("1. " + a.equals(b));
System.out.println("2. " + (a == b));

System.out.println("3. " + a.equals(c));
System.out.println("4. " + (a == c));

System.out.println("5. " + b.equals(c));
System.out.println("6. " + (b == c));
```

1. false
2. false
3. false
4. false
5. true
6. false

(2 marks)

e) What is printed by the following?

```
int a = 5;
int b = 10;

if ((a < b) && !(b < 10)) {
    System.out.println("A");
} else {
    System.out.println("B");
}
```

A

(2 marks)

Question/Answer Sheet

ID:

f) The `positive()` method is *called* in the following way:

```
boolean[] bs = {true, true, false, false};  
double nums = positive(bs, "10");
```

Complete the method header for the `positive()` method (i.e. complete the first line of the method definition).

```
private double positive ( boolean[] nums, String a ) {  
  
    int x = Integer.parseInt(a);  
    double d = 0.0;  
    for (int i = 0; i < x; i++) {  
        if (nums[i]) {  
            d = d + 5;  
        }  
    }  
    return d;  
}
```

(2 marks)

g) What is the output of the following code?

```
String a = "100";  
  
System.out.println(a + 1);  
System.out.println(Integer.parseInt(a) + 1);
```

```
1001  
101
```

(2 marks)

Question/Answer Sheet

ID:

h) What is printed by the following code segment?

```
String[] s = { "zero", "one", "two" };  
  
s[2] = s[1];  
s[1] = s[0];  
  
for(int i = 0; i < s.length; i++) {  
    System.out.print(s[i] + " ");  
}
```

```
zero zero one
```

(2 marks)

i) Write a Java statement which declares and constructs an array of 30 Rectangle objects. The array should be called boxes.

```
Rectangle[] boxes = new Rectangle[30];
```

(2 marks)

j) Assume that the array of Rectangles, called boxes, from the previous question has been created and initialised, and contains no elements which are null. Write a Java statement which prints out the x value of the first element in the array.

```
System.out.println(boxes[0].x);
```

(2 marks)

Question/Answer Sheet

ID:

Question 2 (8 marks)

Consider the following code segment:

```
String word1 = "banana";
String word2 = "compsci";
String word3 = "aaaxaaa";

String result1 = removeAllAs(word1);
String result2 = removeAllAs(word2);
String result3 = removeAllAs(word3);

System.out.println(result1);
System.out.println(result2);
System.out.println(result3);
```

The output of this code is:

```
bnn
compsci
x
```

The `removeAllAs()` method is passed a `String` as a parameter, and returns a new `String` which is the same as the original except all of the lower case 'a' characters have been removed.

The following list of 12 statements is broken into 6 groups of 2 (i.e. 6 pairs). The correct definition of the `removeAllAs()` method can be formed using exactly ONE statement from each pair. The pairs themselves are given in a random order, so you will need to select the correct statement from each pair as well as determine the correct order for the statements.

```
return result;
return word;

String result = "";
String result;

if (word.charAt(i) == 'a')
if (word.charAt(i) != 'a')

for (int i = 0; i < word.length(); i++)
for (int i = 0; i < word.length; i++)

result = result + word.charAt(i);
result = word.charAt(i);

private String removeAllAs(String word)
private String removeAllAs(word)
```

Question/Answer Sheet

ID:

In the space provided below, give the correct definition of the `removeAllAs()` method. You must **only use the statements that were listed on the previous page**, but arranged in the correct order.

You will need to put **all of the opening and closing braces** into the code, and you must **indent** the code correctly.

```
private String removeAllAs(String word) {  
    String result = "";  
    for (int i = 0; i < word.length(); i++) {  
        if (word.charAt(i) != 'a')  
            result = result + word.charAt(i);  
    }  
    return result;  
}
```

(8 marks)

Question/Answer Sheet

ID:

Question 3 (8 marks)

a) What is the output of the following program?

```
public void start() {  
    int[] numbers = {17, 2, 9, 8, 11, 6, 1, 10, 5};  
    doSomething(numbers);  
    for (int i = 0; i < numbers.length; i++) {  
        System.out.print(numbers[i] + " ");  
    }  
}  
  
private void doSomething(int[] nums) {  
    int temp;  
    for (int i = 0; i < nums.length/2; i++) {  
        temp = nums[i];  
        nums[i] = nums[nums.length-1-i];  
        nums[nums.length-1-i] = temp;  
    }  
}
```

5 10 1 6 11 8 9 2 17

(5 marks)

b) Explain, as a summary in your own words, what the doSomething() method does. Your explanation should consist of a SINGLE SENTENCE in English of no more than approximately 10 words.

The doSomething() method reverses the order of the elements in the array

(3 marks)

Question/Answer Sheet

ID:

Question 4 (6 marks)

Complete the output when the following code is executed.

```
Point pt1, pt2, pt3;

pt1 = new Point(25, 35);
pt2 = new Point(25, 35);

System.out.println( "1.  " + (pt1 == pt2) );

pt1.move(5, 10);

System.out.println( "2.  " + pt1.x + "," + pt1.y );

pt2.translate(5, 20);

System.out.println( "3.  " + pt2.x + "," + pt2.y );

pt2 = new Point(25, 35);
pt3 = pt2;

pt2.x = pt3.y;
pt2.y = pt2.x;

System.out.println( "4.  " + pt2.x + "," + pt2.y );
```

| |
|---|
| <pre>1. false 2. 5,10 3. 30,55 4. 35,35</pre> |
|---|

(6 marks)

Question/Answer Sheet

ID:

Question 5 (6 marks)

Draw the output produced by the following code in the grid given at the bottom of the page.

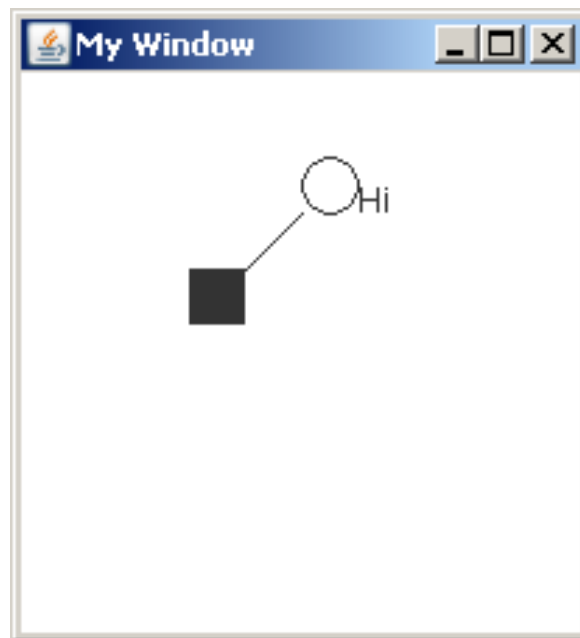
The grid lines are not part of the output but are there to help you place the drawing in the correct position. The size of each square in the grid is 10 pixels by 10 pixels.

```
import java.awt.*;
import javax.swing.*;

public class MyJPanel extends JPanel {

    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        draw(g, 60, 70, 20);
    }

    private void draw(Graphics g, int x, int y, int size) {
        g.fillRect(x, y, size, size);
        g.drawLine(x + size, y, x + size * 2, y - size);
        g.drawOval(x + size * 2, y - size*2, size, size);
        g.drawString("Hi", x + size * 3, y - size);
    }
}
```



(6 marks)

Question/Answer Sheet

ID:

Question 6 (10 marks)

The JPanel defined on the next page contains two JLabel components, two JTextField components and two JButton components:

- numberL – a JLabel displaying "Number"
- numberT – a JTextField which contains a whole number representing the number to be added
- totalSoFarL – a JLabel displaying "Total So Far"
- totalSoFarT – a JTextField which contains the total of all the numbers added so far
- addB – a JButton displaying the String, "ADD TO TOTAL".
- clearB – a JButton displaying the String, "CLEAR".

Below is a screenshot of the JPanel when it is first displayed. The numberT textField displays "0", the totalSoFarT textField displays "0".

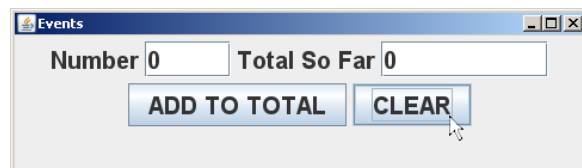
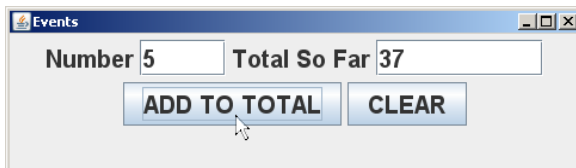
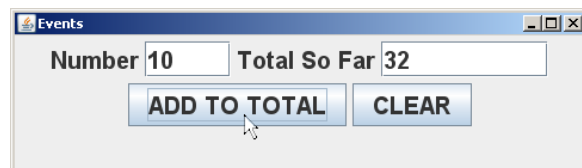
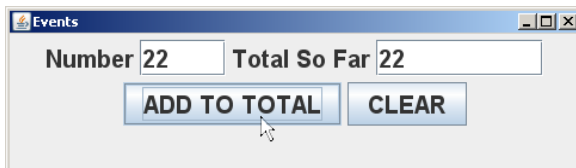


In the numberT textField, the user enters a whole number, representing the value which is to be added to the total displayed in the totalSoFarT textField. When the user presses the "ADD TO TOTAL" JButton, the number from the numberT textField is added to the value in the totalSoFarT textField, and the total is displayed in the totalSoFarT textField.

When the user presses the "CLEAR" JButton, the numberT textField displays "0", the totalSoFarT textField displays "0".

You are required to complete the JPanel definition on the next page so that the JPanel behaves as described above. You must use the variables given in the code.

The first three screenshots below show the JPanel as the user adds three numbers to the total. The last screenshot is displayed after the user presses the "CLEAR" button.



Question/Answer Sheet

ID:

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
import javax.swing.*;
```

```
public class JPanel extends JPanel implements ActionListener {
```

```
    private JTextField numberT, totalSoFarT;  
    private JButton addB, clearB;
```

```
    public JPanel() {
```

```
        JLabel numberL, totalSoFarL;  
        numberL = new JLabel("Number");  
        totalSoFarL = new JLabel("Total So Far");  
        numberT = new JTextField(4);  
        totalSoFarT = new JTextField(8);
```

```
        addB = new JButton("ADD TO TOTAL");  
        clearB = new JButton("CLEAR");
```

```
        addB.addActionListener(this);  
        clearB.addActionListener(this);
```

```
        numberT.setText("0");  
        totalSoFarT.setText("0");
```

```
        add(numberL);  
        add(numberT);  
        add(totalSoFarL);  
        add(totalSoFarT);  
        add(addB);  
        add(clearB);
```

```
    }
```

Question/Answer Sheet

ID:

```
public void actionPerformed(ActionEvent e) {  
    int number, totalSoFar;  
    if (e.getSource() == addB) {  
        number = Integer.parseInt(numberT.getText());  
        totalSoFar =  
            Integer.parseInt(totalSoFarT.getText());  
        totalSoFar = totalSoFar + number;  
        totalSoFarT.setText("" + totalSoFar);  
    } else {  
        numberT.setText("0");  
        totalSoFarT.setText("0");  
    }  
}
```

}

(10 marks)

CONTINUED

Question/Answer Sheet

ID:

Question 7 (6 marks)

Complete the output when the following code is executed.

```
Rectangle rect1, rect2, rect3;  
Point pt1 = new Point(30, 35);  
  
rect1 = new Rectangle(30, 40, 50, 20);  
rect2 = new Rectangle(25, 10, 40, 30);  
  
System.out.println( "1.  " + (rect2.contains(pt1)) );  
  
rect3 = rect2;  
rect2.y = rect2.x;  
  
System.out.println( "2.  " + rect2.x + " " + rect2.y );  
System.out.println( "3.  " + rect3.x + ", " + rect3.y );
```

```
1.  true  
2.  25 25  
3.  25,25
```

(6 marks)

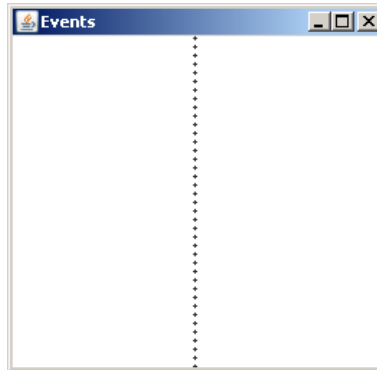
Question/Answer Sheet

ID:

Question 8 (8 marks)

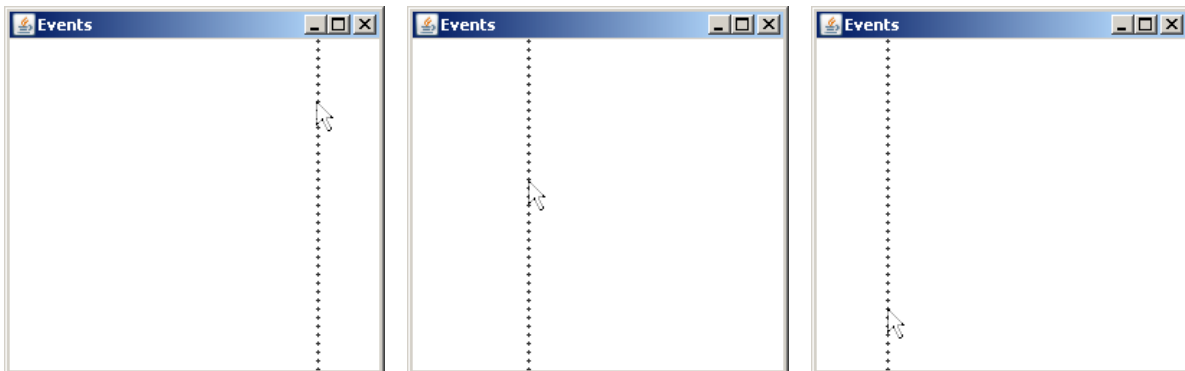
The following JPanel responds to MouseEvents. The JPanel displays a vertical line of dots down the JPanel. Each dot is drawn as an oval which is 3 pixels wide by 3 pixels high.

Below is a screenshot of the JPanel when it is first displayed. Initially, each dot in the vertical line of dots has an x value which is 100.



Whenever the user presses the mouse, the vertical line of dots is displayed at the x value of the position where the user pressed the mouse. The dots are of size 3 pixels and there is a gap of 3 pixels between each dot.

Below are some screenshots of the JPanel in action.



Question/Answer Sheet

ID:

The following numbered statements are part of the code. Place the correct number in the correct place in the `MyJPanel` definition on the next page so that the `MyJPanel` executes as described above.

1. `public void mouseExited(MouseEvent e) {}`
2. `xPos = e.getX();`
3. `int y = 0;`
4. `Graphics g`
5. `for (int i = 0; i < NUMBER_OF_DOTS; i++) {
 g.fillOval(xPos, y, 3, 3);
 y = y + 6;
}`
6. `repaint();`
7. `private int xPos;`
8. `addMouseListener(this);`
9. `implements MouseListener`
10. `xPos = 100;`

Question/Answer Sheet

ID:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class MyJPanel extends JPanel          9      {
    public static final int NUMBER_OF_DOTS = 300;
    7
    public MyJPanel() {
        8
        12
    }
    public void mousePressed(MouseEvent e) {
        2
        6
    }
    public void paintComponent(          4      ) {
        super.paintComponent( g );
        3
        5
    }
    1
    public void mouseClicked(MouseEvent e) {}
    public void mouseEntered(MouseEvent e) {}
    public void mouseReleased(MouseEvent e) {}
}
```

(8 marks)

Question/Answer Sheet

ID:

Question 9 (10 marks)

Below is the IceCream class which you need to complete. A program which uses the IceCream class is shown on the next page. The IceCream class represents a flavoured ice cream. The cost of an ice cream is \$1.50 per scoop plus \$0.50 if the ice cream is in a cone.

```
public class IceCream {
    private String flavour;
    private int numberOfScoops;
    private boolean isInACone;

    public IceCream(String flavour, int numberOfScoops,
                    boolean isInACone) {
        this.flavour = flavour;
        this.numberOfScoops = numberOfScoops;
        this.isInACone = isInACone;
    }

    public void setIsInACone(boolean isInACone) {
        this.isInACone = isInACone;
    }

    public boolean getIsInACone () {
        return isInACone;
    }

    public double getCost() {
        double cost = numberOfScoops * 1.5;

        if (isInACone) {
            cost = cost + 0.50;
        }
        return cost;
    }

    public boolean costsMoreThan(IceCream other) {
        if (getCost() > other.getCost()) {
            return true;
        }

        return false;
    }
}
```

Question/Answer Sheet

ID:

```
public String toString() {
    String iceCreamInfo = flavour + " ice cream";
    iceCreamInfo = iceCreamInfo + ", scoops: " +
        numberOfScoops;

    if (isInACone) {
        iceCreamInfo = iceCreamInfo + ", in a cone";
    }

    return iceCreamInfo;
}
}
```

The code shown below uses the IceCream class.

```
IceCream iceCream1, iceCream2;
iceCream1 = new IceCream( "Berry", 3, true);
iceCream2 = new IceCream( "Chocolate", 1, false);
System.out.println( "1. " + iceCream1.toString() );
System.out.println( "2. " + iceCream2.toString() );

iceCream1.setIsInACone(false);
iceCream2.setIsInACone(true);
System.out.println( "3. " + iceCream1.toString() );
System.out.println( "4. " + iceCream2.toString() );

System.out.println( "5. $" + iceCream1.getCost() );
System.out.println( "6. $" + iceCream2.getCost() );

if (iceCream1.costsMoreThan(iceCream2) ) {
    System.out.println( "7. IceCream1 costs more than
        iceCream2" );
} else {
    System.out.println( "7. IceCream1 does not cost more than
        iceCream2" );
}
```

Complete the IceCream class definition so that the code above compiles and gives the output shown below:

1. Berry ice cream, scoops: 3, in a cone
2. Chocolate ice cream, scoops: 1
3. Berry ice cream, scoops: 3
4. Chocolate ice cream, scoops: 1, in a cone
5. \$4.5
6. \$2.0
7. IceCream1 costs more than iceCream2

(10 marks)

Question/Answer Sheet

ID:

Question 10 (8 marks)

For this question you need to write the `getFirstLetterString()` method. This method is passed one parameter, an array of `String`s. The method returns a `String` containing the first letter of each `String` in the array, i.e., the first letter of the first `String` in the array is concatenated to the first letter of the second `String` in the array, and so on.

Consider the code segment below:

```
String[] words = { "tip", "angle", "net", "glad", "ore" };
String firstLetters = getFirstLetterString(words);
System.out.println(firstLetters);
```

If you have defined the `getFirstLetterString()` method correctly, the output from the above code segment should be:

tango

You can assume that no element in the array passed to the `getFirstLetterString()` is `null`. You can also assume that all the elements in the array contain *at least* one character.

Define the `getFirstLetterString()` method in the space provided below:

```
private String getFirstLetterString( String[] words ) {
    char c;
    String firstLetters = "";
    for (int i = 0; i < words.length; i++) {
        c = words[i].charAt(0);
        firstLetters = firstLetters + c;
    }
    return firstLetters;
}
```

(8 marks)

Question/Answer Sheet

ID:

Question 11 (10 marks)

The following `JPanel` uses a `Timer` object. The `Timer` object is created with a delay of 100 milliseconds. The `JPanel` displays a square box which, once the `Timer` has started, moves continuously down the `JPanel`.

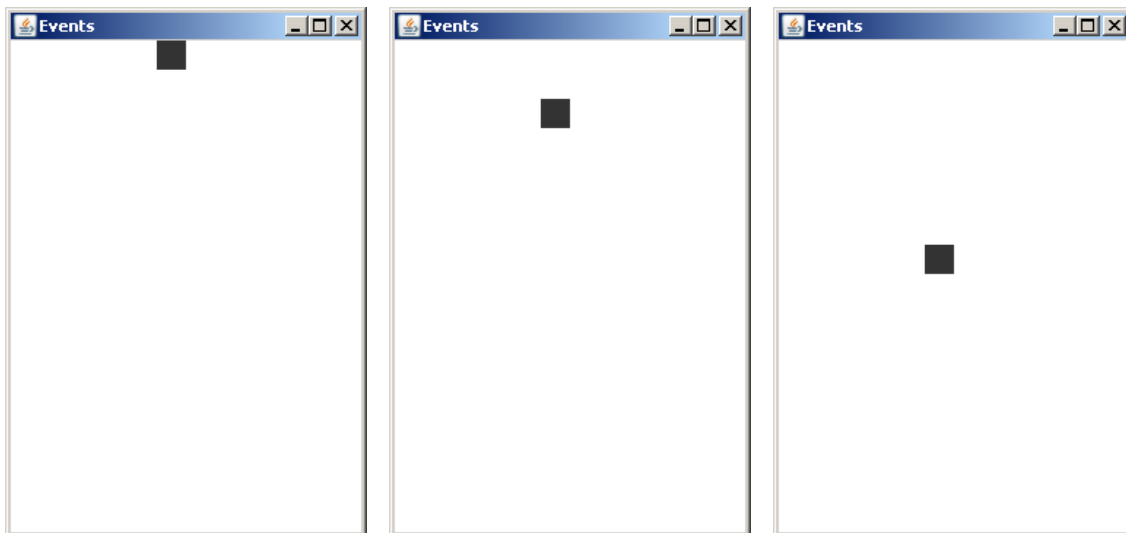
Whenever the user presses the `DOWN` arrow key the `Timer` should start and the box should begin to move from its current position down the `JPanel`. Whenever the user presses the `UP` arrow key the `Timer` should stop and the box should stop in its current position.

Below are some screenshots of the `JPanel` in action. Screenshot 1 shows the `JPanel` when it is first displayed. The box is not moving.

Screenshots 2 and 3 show the `JPanel` as the box moves down the `JPanel`.

NOTES:

- 1) In your code you **MUST** use the instance variables which have been defined. You should not define any other instance variables.
- 2) Each time the box moves, its `y` value is increased by 20 pixels.



Question/Answer Sheet

ID:

```
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;
```

```
public class AJPanels extends JPanel implements ActionListener,  
                                         KeyListener {
```

```
    public static final int SIZE = 20;
```

```
    private Rectangle box;  
    private Timer t;
```

```
    public AJPanels() {
```

```
        box = new Rectangle(100, 0, SIZE, SIZE);
```

```
        t = new Timer(100, this);  
        addKeyListener(this);
```

```
    }
```

```
    public void actionPerformed(ActionEvent e) {
```

```
        box.y = box.y + SIZE;  
        repaint();
```

```
    }
```

```
    public void paintComponent(Graphics g) {  
        super.paintComponent(g);
```

```
        g.fillRect(box.x, box.y, SIZE, SIZE);
```

```
    }
```

Question/Answer Sheet

ID:

```
public void keyPressed(KeyEvent e) {
```

```
    if ( e.getKeyCode() == KeyEvent.VK_UP ) {
        t.stop();
    } else if ( e.getKeyCode() == KeyEvent.VK_DOWN ) {
        t.start();
    }
}
```

```
}
```

```
public void keyReleased(KeyEvent e) {}
```

```
public void keyTyped(KeyEvent e) {}
```

```
}
```

(10 marks)

Question/Answer Sheet

ID:

OVERFLOW PAGE

(If you have used this page, please indicate clearly under the relevant question that you have overflowed to this page)

Question/Answer Sheet

ID:

ROUGH WORKING (WILL NOT BE MARKED)

(You may use this page for rough working)

Question/Answer Sheet

ID:

ROUGH WORKING (WILL NOT BE MARKED)
(You may use this page for rough working)

