THE UNIVERSITY OF AUCKLAND

SECOND SEMESTER, 2008

Campus: City

COMPUTER SCIENCE

TEST

Principles of Programming

(Time allowed: 75 MINUTES)

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:	

SURNAME: ..... FORENAMES: .....

CompSci 101 Test Results

Question	Marks	Out of
Question 1		20
Question 2		10
Question 3		15
Question 4		20
Question 5		20
Question 6		15
TOTAL		100

SURNAME: ..... FORENAMES: .....

Question 1 (20 marks)

What is displayed by each of the following pieces of Java program?

a)
```
System.out.println("\\/\\/");
```
(2 marks)

b)
```
System.out.println(3 + Integer.parseInt("2") * 3);
```
(2 marks)

c)
```
int value = 500;
// no calculator needed for this!
System.out.println( (value / 456)*456 + (value % 456) );
```
(2 marks)

d)
```
int i = 2;
int[] numbers = {4, 2, -7, 5, 1, 6, 3};
System.out.println (numbers[i+1]);
```
(2 marks)

e)
```
String name = "bmi";
System.out.println(name.toUpperCase());
```
(2 marks)

f)
```
String name = "plectrum";
System.out.println(name.substring(0, 2)+ name.substring(6));
```
(2 marks)

g)
```
String word = "supercalifragilisticexpialidotious";
String remark = "amusing";
if (word.length()>20) {
     remark = "sesquipedalian";
}
System.out.println ("word is " + remark);
```
(2 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

h)
```
int x = 5;
double y = 4.0;
System.out.println(Math.max(x, y)); // be careful!
```

(2 marks)

i)
```
int[] numbers = {2,3,4,5,6,7,8};
System.out.println(numbers[4]/numbers[1]);
```

(2 marks)

j)
```
String[] strings = {"panther","jaguar","ocelot","ounce"};
System.out.println(strings[3].length());
```

(2 marks)

Question 2 (10 marks)

Add the return type to the following Java methods:

a)
```
private _____ calculateMinutes(double hours ) {
     return (int)(hours * 60);
}
```

(2 marks)

b)
```
private _____ isRepeated(String s) {
     String firstPart = s.substring(0,s.length()/2);
     return s.equals(firstPart+firstPart);
}
```

(2 marks)

c)
```
private _____ getSurname (String name) {
     int firstSpace = name.indexOf(' ');
     return name.substring(firstSpace + 1);
}
```

(2 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

d)

```
private _____ times2(int num) {
     return num * 2.0;
}
```

(2 marks)

e)

```
private _____ printWages(double payRate, int hours){
     double wages = payRate * hours;
     System.out.println("Wages: " + wages);
}
```

(2 marks)

Question 3 (15 marks)

a) Write a Java `boolean` expression which tests if the value of the `double` variable, `number`, is less than 50.0.

(3 marks)

b) Write a Java `boolean` expression which tests if the value of the `double` variables, `d1`, `d2`, both round down (truncate) to the same `int` value.

(3 marks)

c) Write a Java `boolean` expression which tests if there is a character `'$'` in the `String` variable, `word`, at the position given by the `int` variable `i`

(3 marks)

d) Write a Java `boolean` expression which tests if the value of the `String` variable, `name`, has exactly the same characters in the same order as the value of the `String` variable, `alias`.

(3 marks)

e) Write a Java boolean expression which tests whether the value of the `int` variable, `number`, is equal to either `9` or `16`.

(3 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

Question 4 (20 marks)

Complete each of the methods below as specified in the comment preceding each method.

a)

```
// method to find the maximum of four different numbers
private int maximum(int x1, int x2, int x3, int x4){




}
```

(4 marks)

b)

```
// method to decide whether three numbers are in ascending
// order
private boolean areInOrder(int n1, int n2, int n3){




}
```

(4 marks)

c)

```
// method to return the first half of a String
//(assuming that the length of the String is even)
private String firstHalf(String s){




}
```

(4 marks)

d)

```
// method to tell whether two Strings, name1, and name2,
// are exactly the same. The method returns the String
// "yes" if they are the same, but "no" otherwise
private String theSame(String name1, String name2){









}
```

(4 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

e)

```
// method to "print" a line of '*'s, the number of '*'s
// in the line being given by int parameter, n. There
// should be no newline at the end of line.
private void starLine(int n){




}
```

(4 marks)

Question 5 (20 marks)

a) Complete the method `areCloseEnough()` which accepts three `double` parameters, `d1, d2` and `delta` (a small positive number, e.g. 0.0000001). The method returns `true` when the difference between `d1` and `d2` is less than the value of `delta`, but returns `false` otherwise. (Note that `d1` might be greater than or less than `d2` – the `Math.abs()` method might be useful.)

```
// method to find whether two numbers are close to equal
private boolean areCloseEnough
                (double d1, double d2, double delta) {




}
```

(5 marks)

b) Complete the method `displaySolutions()`. The method has three `double` parameters, `a, b` and `c`. The method finds the two solutions to the quadratic equation ax^2+bx+c as the variables `solution1` and `solution2`, then displays the solutions. The solutions have the values $(-b + \sqrt{(b^2-4*a*c)})/2$ and $(-b - \sqrt{(b^2-4*a*c)})/2$. You may assume that both solutions exist. Hint: `Math.sqrt()` should be useful.

```
// method to display both solutions to a quadratic equation
private void displaySolutions(double a, double b, double c) {









     System.out.print("The solutions are: ");
     System.out.println(solution1 + ", " + solution2);
}
```

(5 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

c) Complete the method `rotateRight()` which accepts a `String` parameter `word`, and returns the `String` rotated one character to the right, e.g. `"luck"` gives the result `"kluc"`

```
// method to find a word rotated one character to the right,
// e.g. "luck" gives result "kluc"

private String rotateRight (String word){









}
```

(5 marks)

d) Complete the method `isAllDigits()` which accepts a `String` parameter, `word`, and returns `true` if every character in `word` is a digit (0, 1, 2, 3, 4, 5, 6, 7, 8, or 9), and `false` otherwise. You may use the `boolean` method `Character.isDigit(char c)` that returns `true` if its parameter `c` is a digit and `false` otherwise.

```
// method to see if a word is all digits

private boolean isAllDigits(String word) {
     boolean allSoFar = true; // all characters so
                              // far tested are digits
```

```
}
```

(5 marks)

SURNAME: ..... FORENAMES: .....

Question 6 (15 marks)

a) What is the output when the following `start()` method is executed?

```
// mystery program
public void start() {
     String s1 = "****";
     String s2 = "*  *";
     System.out.println(s1);
     for (int i = 0; i<(s1.length()/2; i++) {
          System.out.println(s2);
     }
     System.out.println(s1);
     for (int i = 0; i<(s1.length()/2; i++) {
          System.out.println(s2);
     }
     System.out.println(s1);
}
```

Show the output here:

(8 marks)

b) What is the output when the following `start()` method is executed?

```
// owls in the summer
public void start() {
     String s = "toohot";
     System.out.println(s + " " + munted(s));
}

// mystery helper method
private String munted(String s) {
     String result = "";
     for (int j=s.length()-1; j>=0; j--) {
          result = result + s.charAt(j);
     }
     return result;
}
```

Show the output here:

(7 marks)

CONTINUED

SURNAME: ..... FORENAMES: .....

ROUGH WORKING (WILL NOT BE MARKED)

(You may detach this page from the answer booklet and use it for rough working)

SURNAME: ..... FORENAMES: .....

ROUGH WORKING (WILL NOT BE MARKED)

(You may detach this page from the answer booklet and use it for rough working)