THE UNIVERSITY OF AUCKLAND

SUMMER SEMESTER, 2005
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 (output)		10
Question 2 (output)		10
Question 3 (keyboard input)		10
Question 4 (conditionals)		10
Question 5 (loops)		10
Question 6 (loops)		10
Question 7 (methods)		10
Question 8 (methods)		10
Question 9 (arrays)		10
Question 10 (desk-checking)		10
TOTAL		**100**

CONTINUED

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

Question 1 (10 marks)

a) What is printed by the following?

```
double a;
a = 1 / 2;
System.out.println( a );
```

```
0.0
```

(2 marks)

b) What is printed by the following?

```
int x = 9 - 8 + 7 / 6 - 5 % (4 + 3) * 2 - 1;
System.out.println( x );
```

```
-9
```

(2 marks)

c) What is printed by the following?

```
System.out.println( "\" + \"\"" );
```

```
" + ""
```

(2 marks)

d) What is printed by the following?

```
double rand = 0.6;
System.out.println( (int)rand * 100 + 1 );
```

```
1
```

(2 marks)

CONTINUED

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

e) What is printed by the following?

```
int b = (int)(2.5 * 2.0) - (int)3.99;
System.out.println( b );
```

```
2
```

(2 marks)

Question 2 (10 marks)

What is the output when the following code is executed?

```
int num1 = 4;
int num2 = 7;

String word1 = new String( "ABRACADABRA" );
String word2 = new String( word1.toUpperCase() );
String word3 = new String( " TO SING  " );

System.out.println("1. " + word1.substring(5, 8));
System.out.println("2. " + word1.indexOf("RA"));
System.out.println("3. " + word1.indexOf('P'));
System.out.println("4. " + word1.charAt(3));

if (word2 == word1)
     System.out.println("5. ==");
else
     System.out.println("5. not ==");

if (word2.equals(word1))
     System.out.println("6. equal");
else
     System.out.println("6. not equal");

System.out.println("7. " + (word1 == word1));

System.out.println("8. " + (num2/num1>=1 || (num2/3.0>2)));
System.out.println("9. " + ( !(num2 != 7)));
System.out.println("10. " + (num2%2==1 && !(num1>3)));
```

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

Show the output here:

```
1. ADA
2. 2
3. -1
4. A
5. not ==
6. equal
7. true
8. true
9. true
10. false
```

(10 marks)

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

Question 3 (10 marks)

For this question, you need to complete the `AverageProgram` class given below. When the `start()` method is executed, the user should be prompted to enter two integers. The program should then calculate and display the average of the two values entered.

Look carefully at the two examples below. If you complete the `start()` method correctly, the output produced should be identical to that shown below:

```
C:\Test> java AverageApplication
Enter first number: 4
Enter second number: 6
Average = 5.0

C:\Test> java AverageApplication
Enter first number: 2
Enter second number: 9
Average = 5.5
```

Complete the `start()` method below. You can use the `Keyboard.readInput()` method for obtaining user input from the keyboard.

```
public class AverageProgram {
    public void start() {
        System.out.print("Enter first number: ");
        int first = Integer.parseInt(Keyboard.readInput());

        System.out.print("Enter second number: ");
        int second = Integer.parseInt(Keyboard.readInput());

        double average = (first + second) / 2.0;

        System.out.println("Average = " + average);
    }
}
```

(10 marks)

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

Question 4 (10 marks)

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

```
public void start () {
     System.out.println("1.");
     doIf(6, 10);

     System.out.println("2.");
     doIf(2, 1);
}

private void doIf(int x, int y) {
     if (x > 4) {
          System.out.println("line 1");
          if (y > 9 && y != 2) {
               System.out.println("line 2");
               x = 15;
          }

          if (y%2 == 1 || y == 3)
               System.out.println("line 3");

          System.out.println("line 4");
     }
     else if (x <= 2 && y <= 9) {
          if (y < x && y != 0) {
               System.out.println("line 5");
               x = 14;
          }
          System.out.println("line 6");
          x = x - 2;
     }

     if (x%2 == 1)
          System.out.println("line 7");
     else
          System.out.println("line 8");
}
```

Show the output here:

```
1.
line 1
line 2
line 4
line 7

2.
line 5
line 6
line 8
```

(10 marks)

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

Question 5 (10 marks)

For this question, you need to complete the `isPrime()` method shown below. This method should return `true` if the value passed to the method as a parameter is a prime number (i.e. a number greater than 1 which can be divided by only 1 and itself without leaving a remainder) and `false` otherwise.

You can assume that the value passed as a parameter to this method will always be greater than 1.

```
private boolean isPrime( int value ) {
```

```
int i = 2;

while ( i < value ) {
     if (value % i == 0) {
          return false;
     }
     i++;
}

return true;
```

(10 marks)

```
}
```

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

Question 6 (10 marks)

For this question, you need to complete the `UniqueRandomProgram` class given below. When the `start()` method is executed, two random integers between 0 and 9 inclusive should be printed out. These two numbers **must not** be the same.

Look carefully at the two examples below. If you complete the `start()` method correctly, the output produced should be identical in format (although, of course, the values may be different) to that shown below:

```
C:\Test> java UniqueRandomApplication
0 and 4

C:\Test> java UniqueRandomApplication
8 and 1
```

Complete the `start()` method below.

```
public class UniqueRandomProgram {
     public void start() {
          int randomOne = (int)(Math.random()*10);
          int randomTwo = (int)(Math.random()*10);

          while (randomTwo == randomOne) {
               randomTwo = (int)(Math.random()*10);
          }

          System.out.println(randomOne + " and " + randomTwo);
     }
}
```

(10 marks)

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

Question 7 (10 marks)

Write a method, `getComment()`, which accepts one `int` parameter and returns a String. The String which is returned by the method depends on the value of the `int` parameter:

- If the value of the parameter is greater than 8, then the String, "`HIGH`", is returned by the method.
- If the value is greater than 6 and less than or equal to 8 then the String, "`MEDIUM`", is returned by the method.
- If the value is greater than 0 and less than or equal to 6 then the String, "`LOW`", is returned by the method.
- In all other cases, the method returns the String, "`NO COMMENT`".

For example, executing the code below with the completed `getComment()` method produces the following output:

```
8: MEDIUM

6: LOW

-2: NO COMMENT

18: HIGH
```

```
String comment = getComment(8);
System.out.println("8: " + comment);
System.out.println();

comment = getComment(6);
System.out.println("6: " + comment);
System.out.println();

comment = getComment(-2);
System.out.println("-2: " + comment);
System.out.println();

comment = getComment(18);
System.out.println("18: " + comment);
System.out.println();
```

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

```
private  String  getComment( int number ){


     if (number > 8) {
       return "HIGH";
     } else if (number > 6) {
       return "MEDIUM";
     } else if (number > 0) {
          return "LOW";
     }

     return "NO COMMENT";









}
```

(10 marks)

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

Question 8 (10 marks)

Write a method, `getUpperLower()`, which accepts one String parameter and one `int` parameter, and returns a String. The method changes the String, which is passed in as a parameter, into a combination of upper case and lower case characters. The number of upper case characters is given by the value of the `int` parameter. The rest of the characters in the String, are to be all lower case characters.

For example, executing the code below with the completed `getUpperLower()` method produces the following output:

```
HanGMAN: HANgman
aMazinGly So: AMAZingly so
over The TOP: OVER THe top
```

```
String word = "HanGMAN";
System.out.println(word + ": " + getUpperLower(word, 3));

word = "aMazinGly So";
System.out.println(word + ": " + getUpperLower(word, 4));

word = "over The TOP";
System.out.println(word + ": " + getUpperLower(word, 7));
```

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

```
private  String  getUpperLower(  String word, int pos  ){


   String upper = word.substring(0, pos);
   upper = upper.toUpperCase();
   String lower = word.substring(pos);
   lower = lower.toLowerCase();

   return upper + lower;









}
```

(10 marks)

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

Question 9 (10 marks)

What is the output of the `start()` method shown below?

```
public void start() {

     int[] a = { 1, 2, 3 };
     int[] b = { 4, 5, 6 };

     int[] c = new int[3];

     for (int i = 0; i < c.length; i++) {
          c[i] = a[i] + b[b.length-1-i];
     }

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

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

Show the output in the box below:

```
7 7 7
```

(10 marks)

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

Question 10 (10 marks)

What is the output when the `start()` method shown below is executed? You may find it useful to use the desk-checking technique covered in lectures. The space on the facing page can be used to show the diagram you used to desk-check the code.

```
public void start() {

     String endS;
     int num1 = method2(24.6);
     System.out.println("num1: " + num1);

     endS = method1(528, 5);

     System.out.println("endS: " + endS);
}

private String method1(int num1, int len) {

     String numS;
     int num2 = method2(num1);
     len = len - 3;

     num2 = num2/100;
     numS = num2 + "000";
     System.out.println("numS: " + numS);

     numS = numS.substring(0, len);
     return numS;
}

private int method2(double num1) {

     int num2 = (int) Math.round(num1);
     num2 = (num2 + 10) / 10;

     System.out.println("num2: " + num2);
     return num2 * 10;
}
```

Show the output below:

```
num2: 3
num1: 30
num2: 53
numS: 5000
endS: 50
```

(10 marks)

CONTINUED

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

Working for Question 10. This is NOT required, although partial credit may be given for working if the solution is incorrect.

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

OVERFLOW PAGE

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

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)