

**UNIVERSITY OF TORONTO**  
**FACULTY OF APPLIED SCIENCE AND ENGINEERING**

**APS105 — Computer Fundamentals**  
**Final Examination — April, 2004**

Examiner: John Carter

Duration: 2.5 h

**Exam Type: A**

This is a "closed book" examination; no aids are allowed.

**Calculator Type: 4**

No calculators are allowed.

All questions are to be answered on the examination paper. If the space provided for a question is insufficient, extra space is provided at the end of the examination. If you use this extra space, please indicate clearly which question(s) you have answered there.

The examination has 13 pages, including this one.

The marks allocated to the questions, out of a total of 135, are shown in the question headings.

You must use the Java programming language to answer programming questions. Unless specifically directed otherwise, you may use any of the methods from the `Math`, `String`, and `In` classes.

Name \_\_\_\_\_

Student Number \_\_\_\_\_ ecf login \_\_\_\_\_

**MARKS**

1	2	3	4	5	6	7	8	9	10	Total
/10	/20	/15	/15	/15	/15	/7	/8	/15	/15	/135

1. [10 Marks]

Circle the correct answer for each of the following statements. Each correct answer is worth one mark. Each incorrect answer will result in a deduction of one mark.

- (a) **True or False:** A `char` value is stored using 16 bits.
- (b) **True or False:** In Java, the expression `5 < n < 2` will have the value `false` for any value of the `int` variable `n`.
- (c) **True or False:** If we fail to write a `toString` method for a class, Java will provide one automatically.
- (d) **True or False:** In Java, one cannot create an array of objects, only an array of references to objects.
- (e) **True or False:** The declaration `int [] list = new int[10];` automatically initializes the elements of `list`.
- (f) **True or False:** Methods that have recursive calls are always slow.
- (g) **True or False:** For a sequential search to work properly, the list must be sorted.
- (h) **True or False:** Multiple constructors for an object can be created provided that they each have a different return type.
- (i) **True or False:** Selection sort is always faster than bubble sort.
- (j) **True or False:** Array indexes in Java must always start at zero.

2. [20 Marks]

Each part of this question is worth two marks.

(a) Evaluate `(int) 2.5 * 4 % 3`

(b) Write a statement that will assign to the `int` variable `x` a random value in the set  $\{5, 10, 15, 20\}$ .

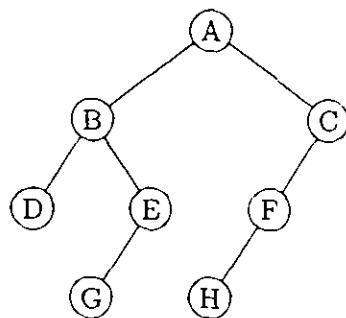
(c) Write a Java expression whose value is  $\sin 10^\circ$ .

(d) What is meant by the *scope* of an identifier?

(e) Trace a binary search as it seeks the value 70 in the array called `list` shown below. To show your trace, print the value of the array element examined at each stage of the search.

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
list	30	35	40	48	56	58	63	66	67	77	80	82	84	89	92

- (f) Suppose that an array initially contains the values {8,2,6,9,5}. If the array is to be sorted into ascending order using insertion sort, show the contents of the array after each of the first two passes of the sort.
- (g) Suppose that an array initially contains the values {5,7,3,2,4}. If the array is to be sorted into ascending order using selection sort, show the contents of the array after each of the first two passes of the sort.
- (h) Suppose that an array initially contains the values {15,23,14,19,11,28,12}. If the array is to be sorted into ascending order using quicksort as discussed in class, show the contents of the array after the first pass of the sort.
- (i) What is the primary difference between an instance field and a class field?
- (j) The diagram shows a binary tree. In what order would the nodes be visited by a postorder traversal?



3. [15 Marks]

Complete the definition of the method `natLog` that computes the value of  $\ln x$  without using the `log` method from the `Math` class. Instead, the method should use the following power series:

$$\ln(x) = \frac{x - 1}{1} - \frac{(x - 1)^2}{2} + \frac{(x - 1)^3}{3} - \frac{(x - 1)^4}{4} + \dots$$

The formula is valid for  $0 < x \leq 2$ . If  $x$  is outside this range, your method should print the message: `argument out of range` and return the value zero. In computing the value of  $\ln x$  for valid values of  $x$ , your method should continue to add terms of the power series until it reaches a term whose absolute value is less than  $10^{-15}$ . Your method should not use any method from the `Math` class other than `Math.abs`.

```
public static double natLog (double x)
```

4. [15 Marks]

A *circle* can be defined as a set of points in a plane that are a given distance (the *radius*) from a given point (the *centre*). Suppose we want to create a class `Circle` whose objects represent circles in the Cartesian plane. Each `Circle` object will have fields:

```
private double x;    // x-coordinate of centre  
private double y;    // y-coordinate of centre  
private double r;    // radius of circle
```

- (a) Write a constructor method for the `Circle` class. The constructor should have three parameters representing the coordinates of the centre and the radius. It should create a new `Circle` object with the given centre and radius.

(b) Write a `toString` method for the `Circle` class. For the circle with centre at (2.5, 3.0) and having radius 6.25, the method should return the string:  
`"circle: centre (2.5,3.0) and radius 6.25"`

(c) Write an instance method `area` that returns the area of its implicit `Circle` object, rounded to two decimal places.



5. [15 Marks]

Complete the definition of the method `indexOf` whose heading is shown below. The method is similar to the `indexOf` method discussed in the text except that the second parameter is a string instead of a character. If `pattern` is a substring of `s`, then `indexOf` should return the index in `s` at which the first occurrence of `pattern` begins. If `s` does not contain `pattern`, then `indexOf` should return `-1`. If either `s` or `pattern` are empty strings, then `indexOf` should return `-1`. You may use any of the methods from the `String` class that are discussed in the text but you may not use any others.

As examples,

```
indexOf("Mississauga", "iss") should return 1  
indexOf("Calgary", "gar") should return 3  
indexOf("Toronto", "orb") should return -1  
indexOf("Ottawa", "") should return -1
```

```
public static int indexOf (String s, String pattern)  
{
```

6. [15 Marks]

Suppose that an array `price` contains a sequence of daily closing prices of a stock. We define a *rally* as a period of time in which the closing price of the stock either increases or stays the same. Complete the definition of the method `bestRally` so that it returns the difference in the price of the stock for the rally having the largest price increase. If the stock price never rises, the method should return the value zero.

```
public static double bestRally (double[] price)
```

7. [7 Marks]

Consider the recursive method whose definition is:

```
public static int f (int n)
{
    int result = 0;
    if (n != 0)
        if (n % 10 == 0)
            result = f(n/10);
        else
            result = 1 + f(n-1);
    return result;
}
```

(a) Evaluate  $f(3)$ .

(b) evaluate  $f(203)$ .

(c) State in a few words (no more than ten) what  $f(n)$  determines.

8. [8 Marks]

Complete the definition of the recursive method `reverse` so that it returns the reversal of its argument.

For example,

```
reverse("Indie")
```

should return the string "eidnI". Your method *must* be recursive. No credit will be given for a non-recursive solution.

```
public static String reverse (String s)
```

9. [15 Marks]

Suppose that linked lists are represented in the usual way seen in class with `List` objects containing a field:

```
private Node head  
and Node objects containing the fields:  
int info;  
Node link;
```

Assuming that lists are maintained in increasing order of `info` values with no repetitions, complete the definition of the method `printUnion` (of the class `List`) so that it prints, in ascending order, the `info` fields that would be contained in the union of the two lists, one value per line, with no repetitions. Your solution should not modify either list, write or call any other methods except `System.out.println`, or create any new objects. (The union of two lists contains the elements that are only in the first list, only in the second list, or in both lists.)

```
public void printUnion (List other)
```

10. [15 Marks]

Complete the definitions of the methods `printRange` whose headers are shown below. The methods should assume that the tree on which they are operating is organized as a binary search tree in which all `info` fields are unique. The methods should print the values of the `info` fields that fall in the range  $\text{low} \leq \text{info} \leq \text{high}$ . Values should be printed one per line. The methods should visit as few nodes as possible.

```
class Tree
{
    private Node root;

    public void printRange (int low, int high)
    {

    }

    class Node
    {
        int info;
        Node lChild;
        Node rChild;

        void printRange(int low, int high)
        {
            if (info < low)
                lChild.printRange(low, high);
            else if (info > high)
                rChild.printRange(low, high);
            else
                System.out.println(info);
            if (lChild != null)
                lChild.printRange(low, high);
            if (rChild != null)
                rChild.printRange(low, high);
        }
    }
}
```

**Extra Space**    *Please specify which question(s) you are answering on this page.*