University of Toronto
Faculty of Applied Sciences and Engineering

Final Examination, April 2007
First Year – Industrial Engineering
APS105H1S-MIE – Computer Fundamentals

Examiner: A. Ghazarian

Duration: 2.5 hours (150 minutes)

Exam Type A:
This is a "closed book" examination; no aids are permitted.

Calculator Type 4:
No electronic or mechanical computer devices are permitted.

Write your answer in the space that follows each question.

If necessary, use the back of the pages for rough work.

This examination has 12 pages.

You must use the Java programming language to answer the programming questions.

Full Name:

Student Number:

MARKS

| Question# | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | TOTAL |
|---|---|---|---|---|---|---|---|---|---|---|
| Mark | | | | | | | | | | |
| Value | /5 | /5 | /5 | /5 | /5 | /5 | /5 | /10 | /5 | /50 |

**Question 1** [5 Marks]
Consider the following class:

```
Public class Book {
     String title;
     String author;
     int numberOfPages;
     double price;
}
```

The following methods are supposed to be constructors for this class. For each constructor write:

- **"Correct"** if the constructor is correct (is a valid constructor for the class), or
- **"Wrong"** if the constructor is not correct (has programming error(s))

For each wrong constructor, explain the reason why the constructor is wrong, and write the correct version of the constructor.

a)

```
public int Book(int numPages) {
     numberOfPages = numPages;
}
```

b)

```
public Book (String title) {
     title = title;
}
```

c)

```
public Book (String t, String a, int n, double p ) {
     title = t;
     author = a;
     numberOfPages = n;
     price = p;
}
```

d)

```
public book (String author) {
     this.author = author;
}
```

e)

```
public Book(int numPages, int numChapters) {
     this.numberOfPages = numPages;
     this.numberOfChapters = numChapters;
}
```

**Question 2** [5 marks]
Complete the method below so that it returns the maximum value found in `values`, an array of float numbers passed as a parameter. You may assume that there is at least one item in the array.

```
Public static float max(float[] values) {




}
```

Question 3 [5 Marks]
Complete the method below so that it returns the string s with the characters reversed. As an example, if we pass the string "computer" as a parameter to the method, it should return the string "retupmoc".

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




}
```

Question 4 [5 marks]
Complete the following method:

```
/** Given a 2D array of integers, return a 1D array, whose elements contain
 *  the averages of their corresponding rows in the 2D array. This means
 *  that the first element in the returned 1D array should contain the average
 *  of the numbers in the first row of the given 2D array. Similarly, the
 *  second element in the returned 1D array should contain the average of
 *  the numbers in the second row of the given 2D array, and so on.

Public static int[] computeAverageForEachRow(int[][] a) {




}
```

Question 5 [5 marks]
Complete the following method:

```
/** "bookList" contains a comma-separated String of book information in
 * this form:  bookName#author#price
 * This means that books are separated using ',', whereas each piece of
 * information within a book is separated using a '#' character.
 * Here is an example:  "Java 2#Schildt#40,APS105 Text#Carter#30"
 * This method prints a header for each book (e.g., Book 1, Book 2, and so
 * on)  and prints each piece of information (book name, author name, and
 * the  price) in a separate line.
 * Using our example, the following should be printed:
 * Book 1
 * Java 2
 * Schildt
 *  40
 * Book 2
 * APS105 Text
 * Carter
 * 30
 */
public static void printBookInfo(String bookList){









}
```

Question 6 [5 marks – 2.5 marks each]
Suppose an array initially contains the values {31,89,64,37}. We want to sort this array from smallest to largest.

a) Show the contents of the array after each of the passes (at the end of each complete set of iterations of the inner loop in the algorithm) of an insertion sort algorithm.

b) Show the contents of the array after each of the passes (at the end of each complete set of iterations of the inner loop in the algorithm) of a selection sort algorithm.

Question 7 [5 marks]
Complete the method below. The method should check the matrix to see if the numbers in both diagonals all have the same value. For example, the method should return true for the following matrix because all of the elements on both of its diagonals have a value of 5.

| | | | |
|---|---|---|---|
| 5 | 1 | 0 | 5 |
| 4 | 5 | 5 | 3 |
| 7 | 5 | 5 | 2 |
| 5 | 2 | 1 | 5 |

```
/** Returns true if all of the values on both diagonals in the square array a
 *  have the same value, and false otherwise.
 */
public static Boolean sameValueOnDiagonals(int[][] a) {




}
```

Question 8 [10 marks]

a) Complete the following code based on the comments provided. [4 marks]

```
/** An Employee with a name, department, employee
 *  number, and an hourly pay rate.
 */
public class Employee {

    //Declare attributes (variables) here.




    /** Creates a new Employee with name n, department d,
    * employee number n, and hourly pay rate h.
    */




    /** Get the name of this employee. */
```

```
    /** Takes the number of hours worked as an int
     *   parameter and calculates and returns back the
     *   employee's salary.
     *   The salary is calculated by first multiplying the
     *   number of hours worked by the hourly pay rate and
     *   then subtracting a 10 percent tax from the salary.
     */
```

```
}
```

b) This question continues from part a. Complete the following code based on the comments provided. [6 marks]

```
/** A company with a name and a list of Employee objects.
*/
public class Company {

   //Declare any necessary attributes (variables) here.
```

```
/** Creates a new company with name n, and a maximum
 * number of p employees.
 */
```

```
/** Takes an object of type Employee as a parameter and
 *  if there is still room in the company, adds the
 *  employee to the company, and returns true
 * (successful). If the employee cannot be added to the
 * company (not enough space) returns false.
 */
```

```
/** Assuming that each employee works exactly 160 hours per
 *  month, prints a list of employee names and their
 *  monthly salaries.
 */
```

```
}
```

Question 9 [5 marks]
Assume that each node in a linked list is represented by the class Node provided below.

```
Public class Node {
     Public int info;
     Public Node link;
}
```

Also assume that the class that contains the "deleteItem" method has an attribute called "head" that points to the first element in the linked list (the front of the list). Note that if the list is empty, then the "head" is null. With these assumptions in mind, complete the "deleteItem" method below.

```
/** deletes the item from the linked list, if it can be
 *  found in the list.
 */
Public void deleteItem(int item) {









}
```