ComS 207: Programming I
Midterm 2, Tue. Oct 23, 2007

Student Name:
Student ID Number:
Recitation Section:

1. **True/False Questions (10 x 1p each = 10p)**

    (a) I forgot to write down my name and student ID number.   TRUE / FALSE

    (b) This is an infinite loop: while((!b && c) || true) k++;   TRUE / FALSE

    (c) This is an infinite loop: for(int i=0; i< 100; j++); i++;   TRUE / FALSE

    (d) In a Java program only one object may have a main method.   TRUE / FALSE

    (e) An array index cannot be negative.   TRUE / FALSE

    (f) In a 2D array the second dimension is encoded with 1D arrays.   TRUE / FALSE

    (g) A method can have two variable length parameter lists.   TRUE / FALSE

    (h) Java methods can accept 2D arrays as arguments.   TRUE / FALSE

    (i) The following statement is: (!(a || b) != (!a && !b))   TRUE / FALSE

    (j) The following statement is: (!(a && b) == (!a || !b))   TRUE / FALSE

2. **Short Answer Questions (5 x 2p each = 10p)**

(a) What is a private method?

(b) What is defined by this line of Java code: `float[][] n = new float[5][];`

(c) What is the difference between these two statements: `for(;;);` and `while(false);`

(d) What is the difference between a case statement and a switch statement?

(e) What is the difference between an int array and an array of Strings?

3. Code Snippets (3 x 5p each = 15p)

Write a code snippet (3-6 lines max) that produces the result specified below.

    (a) Print all 26 letters of the alphabet (lowercase) separated by commas.

    (b) Print the odd numbers between -52 and 40 separated by commas.

    (c) Given a number n print the value of n factorial (n! = 1 * 2 * ... * n).

4. **Rewriting Code (2 x 7.5p each = 15p)**

(a) Rewrite the following code using for loops

```
int count1 = 1;
int iteration = 1;
while(count1 <= 10) {
    int count2 = 1;
    while(count2 <= 20) {
        System.out.println("Iteration" + iteration++);
        count2++;
    }
    count1++;
}
```

(b) Rewrite the following switch statement using only if and else statements.
```
switch(n) {
    case 0: case 1: case 2:
        System.out.println("A");
        break;
    case 3: case 4:
        System.out.println("B");
        break;
    case 7: case 8: case 9:
        System.out.println("C");
        break;
    default:
        System.out.println("D");
}
```

5. **What is the Output? (2 x 7.5p each = 15p)**

For each of the following code snippets write down what will be printed on the screen.

(a)
```
int n=2;
for(int a=-n; a<=n; a++) {
    for(int b=-n; b<=n; b++)
        if(Math.abs(a) + Math.abs(b) <= n)
            System.out.print("#");
        else
            System.out.print(" ");
        System.out.println();
}
```

(b)
```
for(int a=-1; a<=1; a++) {
    for(int b=-1; b<=1; b++)
        if(Math.abs(a) <= Math.abs(b))
            System.out.print("#");
        else
            System.out.print(" ");
        System.out.println();
}
```

6. **Programming Projects (TOTAL 75p, but each has a different weight)**

    (a) Numbers (15p) Write a complete Java program which prompts the user for a nonnegative value n. The program should then print the following output:

    ```
    1 2 3 ... n-1 n
    1 2 3 ... n-1
    ...
    1 2 3
    1 2
    1
    ```

(b) Daily Calendar (15p)

Write a complete Java program which uses for loops to print a daily calendar of the form given below.

```
 9:00 a.m.
 9:15 a.m.
 9:30 a.m.
 9:45 a.m.
10:00 a.m.
10:15 a.m.
10:30 a.m.
10:45 a.m.
   ...
 5:00 p.m.
 5:15 p.m.
 5:30 p.m.
 5:45 p.m.
 6:00 p.m.
```

(c) Monotonic Sequence (15p)

A sequence of numbers is monotonically increasing if the values in the sequence are sorted in oder. For example, 1, 2, 2, 4, 6 is a monotonic sequence but 1, 4, 3, 5, 7 is not because 4 is greater than 3. Write a complete Java program which tests if a sequence of numbers is monotonically increasing. You can assume that the sequence is stored in an int array (e.g., int[ ] seq = {1, 4, 3, 5, 7};).

(d) Random Permutation (15p)

Write a complete Java program which takes an integer array and produces a random permutation of the array. In other words, the program randomly reshuffles the entries of the array (similar to shuffling a deck of cards). The resulting array must be stored in the memory allocated for the original array (i.e., the permutation is done in place).

(e) Tic-Tac-Toe (15p)

Write a complete Java program which inspects the final configuration of a Tic-Tac-Toe board and announces the winner or declares a tie. The program must also print the location of the winning triple (e.g., row 1-3, column 1-3, main diagonal, or minor diagonal). The board is stored in a 2D char array of the form:

```
char[][] board ={{'x', 'o', 'x'},
                 {'x', 'o', 'o'},
                 {'x', 'x', 'o'}};
```

Sample output: Player x wins. See column 1.

That's it. Good Luck!

Question	Max	Score
True/False	10	
Short Answer	10	
Code Snippets	15	
Rewriting Code	15	
What is the Output	15	
Numbers	15	
Daily Calendar	15	
Monotonic Sequence	15	
Random Permutation	15	
Tic-Tac-Toe	15	
TOTAL:	140	