5COSC001W - Tutorial 6 Exercises

1 Two-dimensional Arrays

Given the following classes:

```
import java.util.*;

class Board {
    int ar[][];

    // creates a nxn size board
    public Board(int n) {
        ar = new int[n][n];

        // fill in array with random ints in the range [0, 100]
        Random generator = new Random();
        for (int i=0; i < n; i++)
            for (int j=0; j < n; j++)
                ar[i][j] = generator.nextInt(101);
    }

    // display the contents of the board
    public void print() {
        for (int[] r : ar) { // for all rows
            for (int c : r)  // for every element in current row
                System.out.print(c + " ");
            System.out.println();
        }
    }
}

public class TwoDimensionalTest {
    public static void main(String[] args) {
        Board board1 = new Board(5);
        board1.print();
    }
}
```

Implement methods `findMaxInRow(int row)`, `findMaxInColumn(int column)`, `findMaxInDiagonal()` of class `Board`, which find the maximum value of a given row, a given column and all diagonals respectively.

For example, given the board:

```
3 5 8
1 2 9
4 5 6
```

`findMaxInRow(2)` would return 9 (the max in the second row), `findMaxInColumn(1)` would return 4 (the max in the first column), and `findMaxInDiagonal()` would return 9 (the max value found among all left-to-right diagonals (3, 2, 6), (1, 5), (4), (5, 9) and (8)).

Hint: To find the maximum value found among all diagonals of a 2-dimensional array, consider how many (and which) elements of the array need to be examined. Do you really have to traverse the array in a diagonal manner to determine the maximum value?

2 Manipulating Arrays

1. Guess what will be the output of the following program without running it.

```
public class ArrayManipulator {
    public static void main(String[] args) {
        int[] a = new int[5];

        for (int i=0; i < a.length; i++)
            a[i] = i + 2;

        int b[] = new int[10];
        for (int i=(b.length-1); i >=0; i--)
            b[i] = i - 2;

        int c[][] = new int[2][];
        c[0] = b;
        c[1] = a;

        a = b;

        /* print all the values in a, b, c */
        System.out.print("a contains: ");
        for (int e : a)
            System.out.print(e + " ");
        System.out.println();

        System.out.print("b contains: ");
        for (int e : b)
            System.out.print(e + " ");
        System.out.println();

        System.out.println("c contains: ");
        for (int[] r : c) {
            for (int e : r)
                System.out.print(e + " ");
```

```
                System.out.println();
            }
        }
    }
```

2. Verify your guess by running the program, and justify any discrepancies between the output and your guess.

3 Generics - ArrayLists

Given the following class `Book` containing the name of the author and the title of a book:

```
public class Book {
    public String author;
    public String title;
}
```

1. Implement a class `Library` which has a single field of type ArrayList, being able to store objects of class `Book`.
2. Implement a method `populate()` which populates the `ArrayList` inside `Library` with 4 book objects.
3. Implement a method `displayAllBooks()` of class `Library`, which displays the author names and titles for all the books found in a `Library` object.

4 Non-parameterised ArrayLists

1. What is wrong with the following program, in the lines indicated with the comment "Problematic line"?

```
import java.util.ArrayList;

class Book {
    public String author;
    public String title;
}

public class ContainerTest {
    public static void main(String[] args) {
        ArrayList list = new ArrayList();

        list.add(3.4);
        list.add(new Integer(4));

        Book myBook = new Book();
        list.add(myBook);
```

```
        double d;
        d = list.get(0);  // Problematic line
        Double d2 = (Integer) list.get(1);  // Problematic line
        Book b3 = (Book) list.get(1);  // Problematic line
        Book b2 = list.get(2);  // Problematic line
    }
}
```

2. Fix the problems (and test by recompiling/running), after modifying only the lines indicated with the comment "Problematic line"? No other lines of the source code should be changed.

3. Make sure you identify which problems occur at compile time and which at run time, in the given program.

5 Challenge: Repeating and missing numbers

Given an unsorted array of size n. Array elements are in the range from 1 to n. One number from set $\{1, 2, \ldots n\}$ is missing and one number occurs twice in the array. Find these two numbers.