**Assignment #4** (Due week #5) – Two Programs, Two Java Files to submit.

1. Create a new project in your IDE – call it whatever you like.
2. Create a new package in your project called: **edu.webster.arrays**
3. All classes you create will go in package: **edu.webster. arrays**

**First Program: BoysAndGirlsNames.java**

1. Start with the given code shell and be sure it is in package **edu.webster.arrays.**
2. **Copy the given BoyNames.txt and GirlNames.txt to package edu.webster.arrays when done it should look like this below.**

A screenshot of a computer

Description automatically generated

1. You will fill in code for the two static methods with the signatures given.

private static void printAllNamesFoundInBothLists(ArrayList<String> boys, ArrayList<String> girls)

AND

private static boolean isNameFound(String nameToFind, ArrayList<String> myListOfNames)

1. Once your methods are written your correct output should look like this.

A white screen with black text

Description automatically generated

1. Note: When I grade your program, I will be using my own data sets of names.
2. Don’t forget that the end of your main program the run should have this statement: “The above program was written by: **PUT YOUR NAME HERE”**

**Second Program: ArrayUtilities.java**

1. Start with the given code shell and be sure it is in package **edu.webster.arrays.**
2. Using the code given code three remaining methods. (look for TODOS)
3. private static double getAverageOfArray(int[][] myArray)

private static int getRowTotal(int[][] myArray, int rowNumber )

private static int getColumnTotal(int[][] myArray, int columnNumber)

1. When done your output should look like this when you run main() for the given array.

A white background with black text

Description automatically generated

1. Note: When I grade your program, I will be be using a different array set and ask for different rows/columns.
2. Don’t forget that the end of your main program the run should have this statement: “The above program was written by: **PUT YOUR NAME HERE”**

**Note: Double check that both Java files are submitted.**

Programming Grading Scale:

Accumulative:

10% Code is neatly formatted, meaningful variable/method names and documented with comments (where meaningful).

10% Is not trivial (is a good faith attempt) and compiles.

75% Code runs (100% of instructor test cases) and meets specified criteria.

5% Code ends with your name specified.

For example:  
  
Code submitted on time:          No Penalty applied  
  
**Code is neatly formatted, etc                  +10%**  
  
**Compiles (and is a good faith attempt):       +10%**  
  
**Code runs 2 out of 3 test cases – 2/3’s of 75% +50%**  
  
Paper is **missing** ending name statement:         **+0%**  
  
Final Assignment Score:                         **70%**

Note: The **BOLDED** entries above will be scored based on instructor discretion.