```java

import java.util.Arrays;
import java.util.Scanner;
import java.io.*;

/**
 * Name: Zane Emerick
 * Class: CS 1450
 * Assignment #1
 * Due: Jan 30, 2020
 *
 * Description: Create a program that fills randomly sized arrays
 *  with random numbers. Sort and display both arrays, before
 *  merging and writing them to a file. Finally, read both arrays
 *  from the file and remove duplicate numbers.
 */
public class EmerickZaneAssignment1 {
    public static void main(String[] args) throws IOException{

        //create lengths of arrays and display
        int sizeA = (int)(Math.random() * 10) + 1;
        int sizeB = (int)(Math.random() * 10) + 1;

        System.out.println("Size 1 = " + sizeA);
        System.out.println("Size 2 = " + sizeB);

        //generate and fill arrays
        int[] arrayA = new int[sizeA];
        int[] arrayB = new int[sizeB];

        for (int i = 0; i < arrayA.length; i++) {
            arrayA[i] = (int)(Math.random() * 25) + 1;
        }

        for (int i = 0; i < arrayB.length; i++) {
            arrayB[i] = (int)(Math.random() * 25) + 1;
        }

        //sort then display arrays
        Arrays.sort(arrayA);
        Arrays.sort(arrayB);

        System.out.println("\nThe first array has " + sizeA + " numbers");
        for (int i : arrayA) {
            System.out.println(i);
        }

        System.out.println("\nThe second array has " + sizeB + " numbers");
        for (int i : arrayB) {
            System.out.println(i);
        }


        //start file writing section of assignment
        System.out.println("\nWriting " + (sizeA + sizeB) + " values to the
file:");

        File file = new File("assignment1.txt");
        PrintWriter resultsPrinter = new PrintWriter(file);

```

```
60          //runs while both arrays still have leftover values
61          int countA = 0;
62          int countB = 0;
63          while(countA < sizeA && countB < sizeB) {
64              if(arrayA[countA] <= arrayB[countB]) {
65                  resultsPrinter.println(arrayA[countA]);
66                  System.out.println(arrayA[countA]);
67                  countA++;
68              } else {
69                  resultsPrinter.println(arrayB[countB]);
70                  System.out.println(arrayB[countB]);
71                  countB++;
72              }
73          }
74
75          //runs when one array empties, and finishes the other array
76          while(countA < sizeA) {
77              resultsPrinter.println(arrayA[countA]);
78              System.out.println(arrayA[countA]);
79              countA++;
80          }
81          while(countB < sizeB) {
82              resultsPrinter.println(arrayB[countB]);
83              System.out.println(arrayB[countB]);
84              countB++;
85          }
86          resultsPrinter.close();
87
88
89          //start file reading portion of assignment
90          System.out.println("\nReading from file:");
91          Scanner fileReader = new Scanner(file);
92
93          int[] ReadArrayNoDuplicates = new int[sizeA + sizeB];
94
95          ReadArrayNoDuplicates[0] = fileReader.nextInt();
96
97          //loop through file and save to array if value isn't a duplicate
98          int readCount = 1;
99          do{
100             int currentValue = fileReader.nextInt();
101             if(ReadArrayNoDuplicates[readCount - 1] != currentValue) {
102                 ReadArrayNoDuplicates[readCount] = currentValue;
103                 readCount++;
104             }
105         } while(fileReader.hasNext());
106
107         fileReader.close();
108
109         //copy read array with zeros to a shorter array without them
110         int readArrayLength = 0;
111         for (int i : ReadArrayNoDuplicates) {
112             if(i != 0) {
113                 readArrayLength++;
114             }
115         }
116
117         int[] finalReadArray = new int[readArrayLength];
118
119         for (int i = 0; i < finalReadArray.length; i++) {
```

```
            finalReadArray[i] = ReadArrayNoDuplicates[i];
        }

        //print final values to console
        for (int i : finalReadArray) {
            System.out.println(i);
        }
    }

}
```