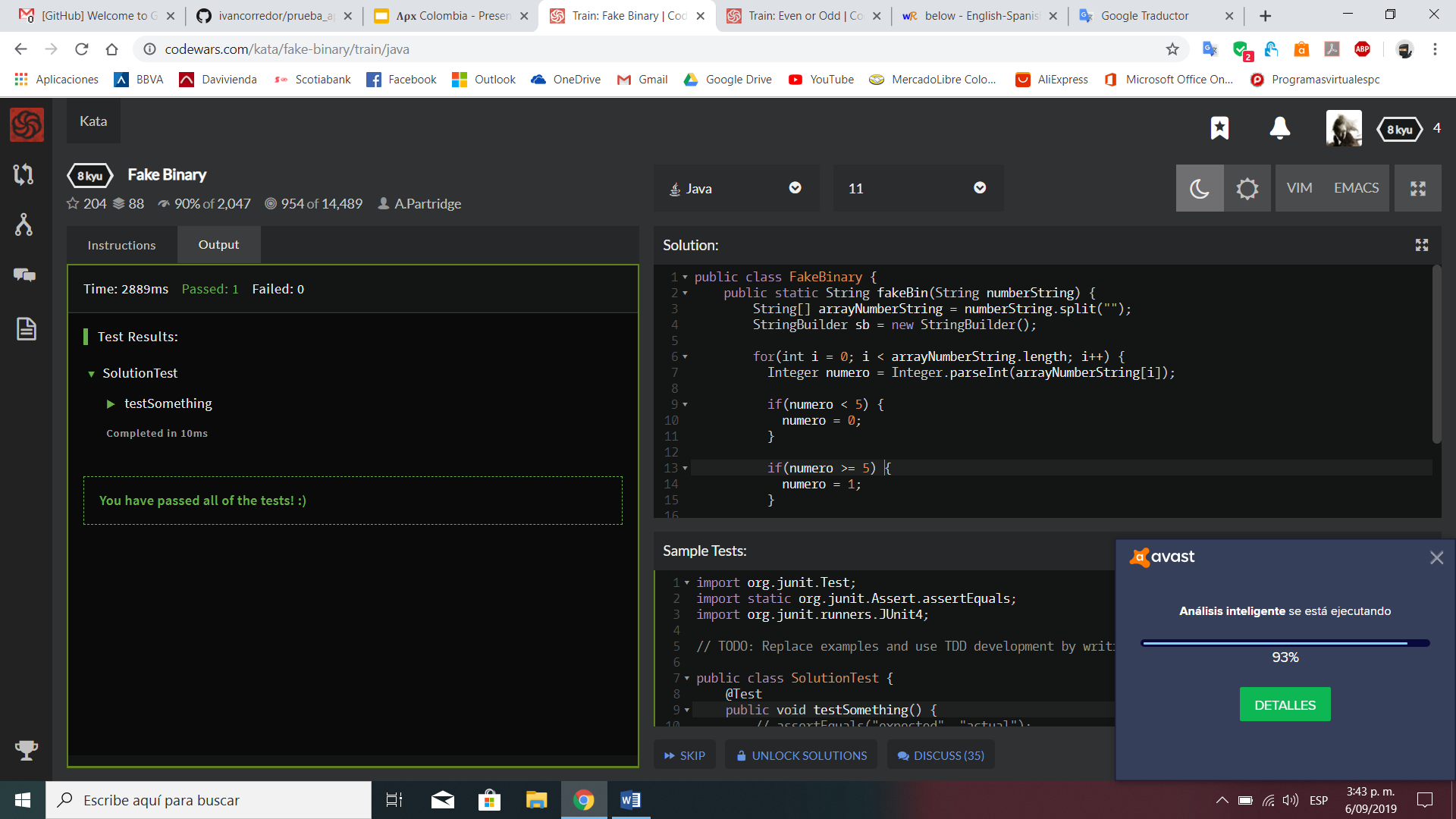
Usuario codewars: **icorredore**

Ejercicio 1



Código fuente:

**public class FakeBinary {**

**public static String fakeBin(String numberString) {**

**String[] arrayNumberString = numberString.split("");**

**StringBuilder sb = new StringBuilder();**

**for(int i = 0; i < arrayNumberString.length; i++) {**

**Integer numero = Integer.parseInt(arrayNumberString[i]);**

**if(numero < 5) {**

**numero = 0;**

**}**

**if(numero >= 5) {**

**numero = 1;**

**}**

**sb.append(numero.toString());**

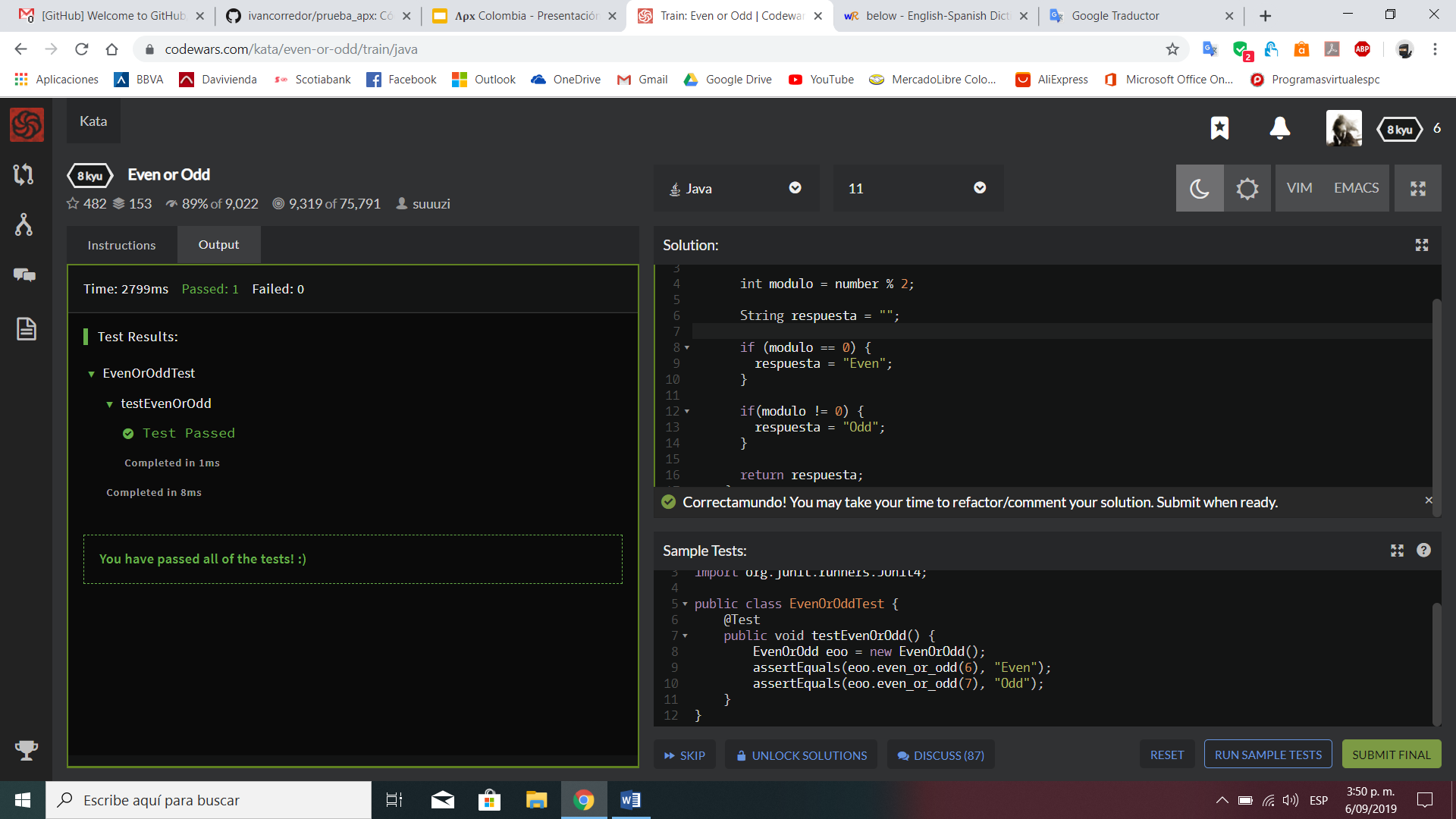
**}**

**return sb.toString();**

**}**

**}**

Ejercicio 2



Código fuente:

**public class EvenOrOdd {**

**public static String even\_or\_odd(int number) {**

**int modulo = number % 2;**

**String respuesta = "";**

**if (modulo == 0) {**

**respuesta = "Even";**

**}**

**if(modulo != 0) {**

**respuesta = "Odd";**

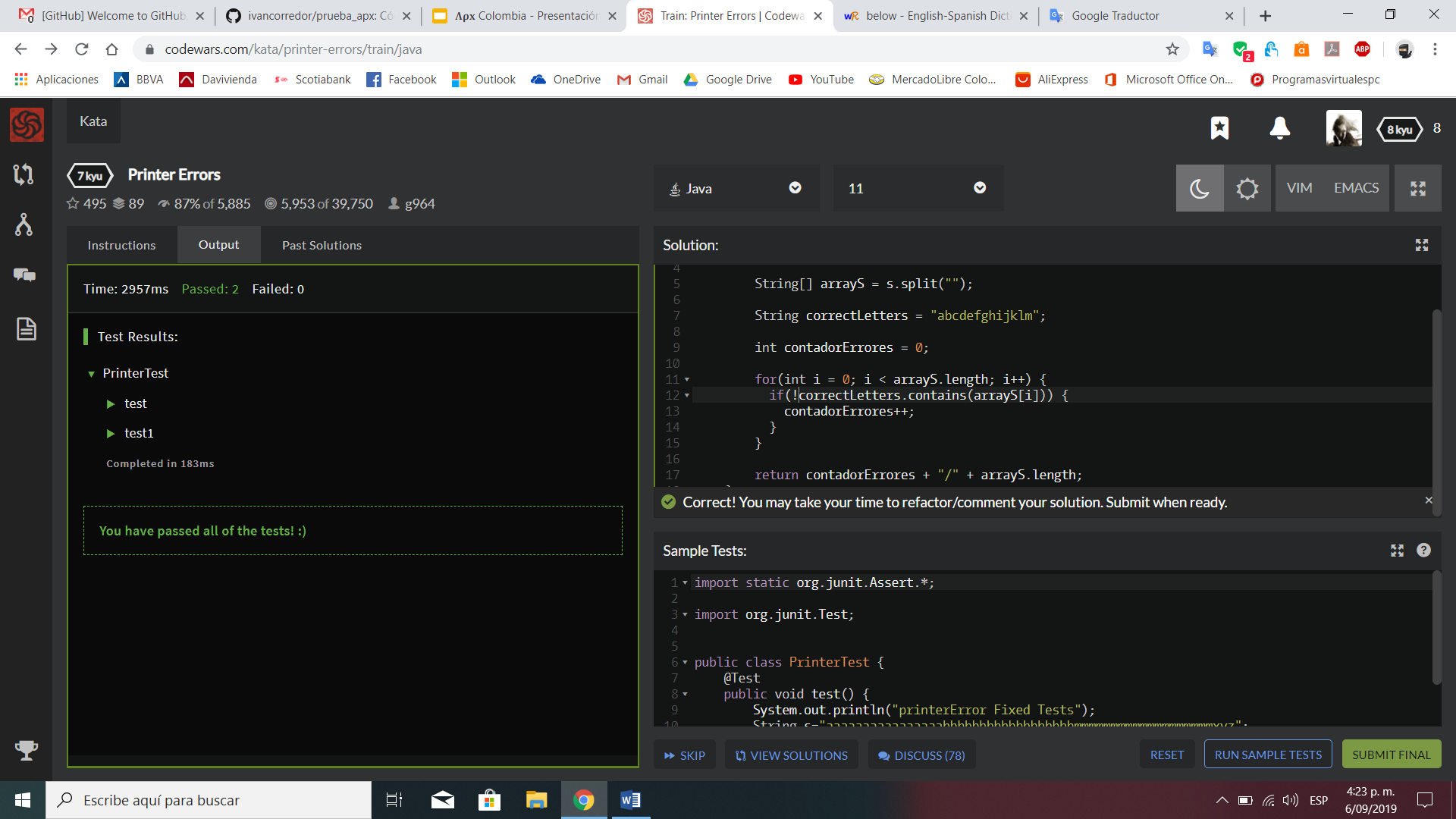
**}**

**return respuesta;**

**}**

**}**

Ejercicio 3



Código Fuente:

**public class Printer {**

**public static String printerError(String s) {**

**String[] arrayS = s.split("");**

**String correctLetters = "abcdefghijklm";**

**int contadorErrores = 0;**

**for(int i = 0; i < arrayS.length; i++) {**

**if(!correctLetters.contains(arrayS[i])) {**

**contadorErrores++;**

**}**

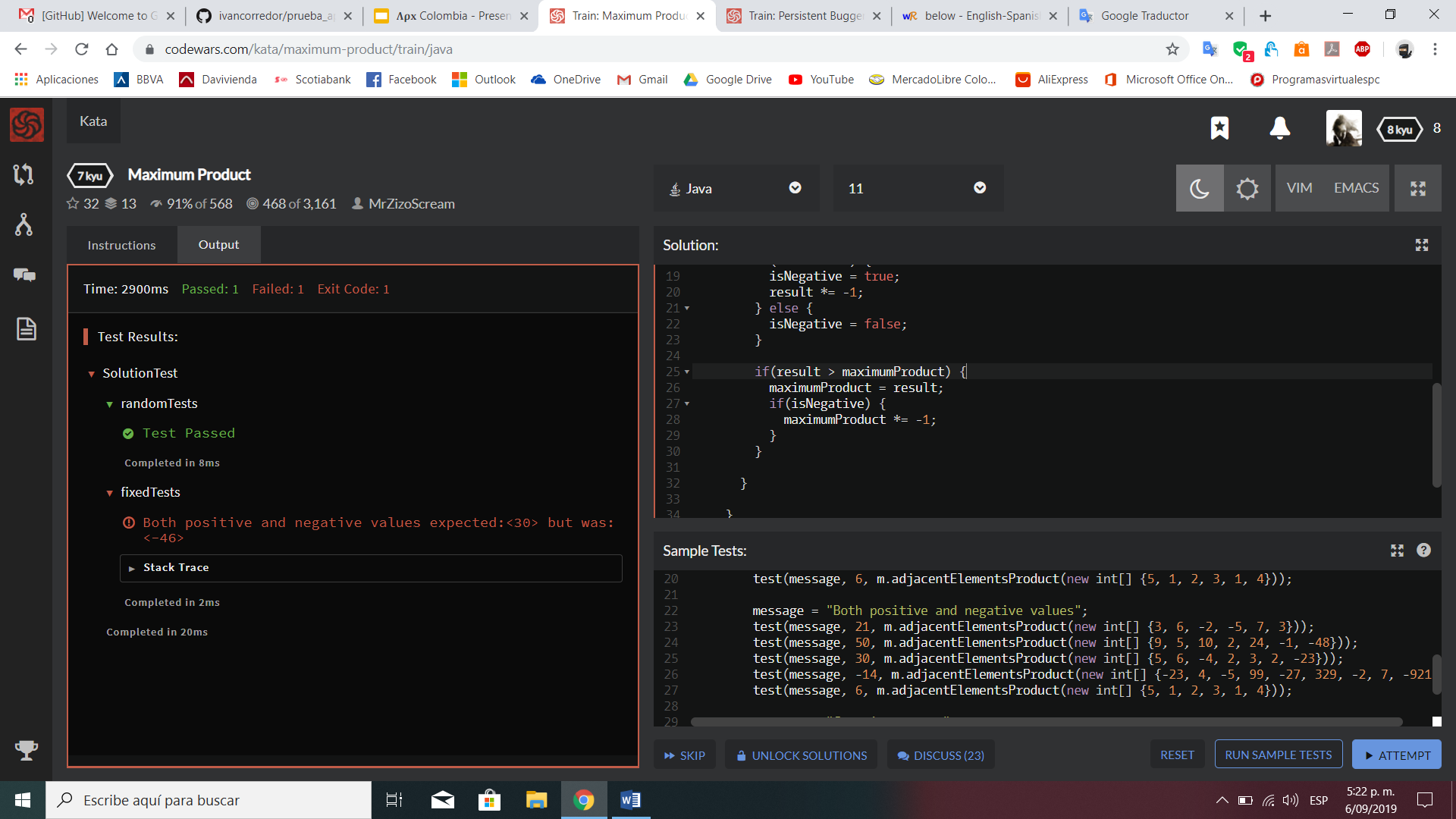
**}**

**return contadorErrores + "/" + arrayS.length;**

**}**

**}**

Ejercicio 4:



Código fuente:

public class MaxProduct {

public int adjacentElementsProduct(int[] array) {

int result = 0;

int maximumProduct = 0;

boolean isNegative = false;

int finalResult = 0;

for(int i = 0; i < array.length; i++) {

if(i < array.length-1) {

result = array[i] \* array[i+1];

if(result < 0) {

isNegative = true;

result \*= -1;

} else {

isNegative = false;

}

if(result > maximumProduct) {

maximumProduct = result;

if(isNegative) {

maximumProduct \*= -1;

}

}

}

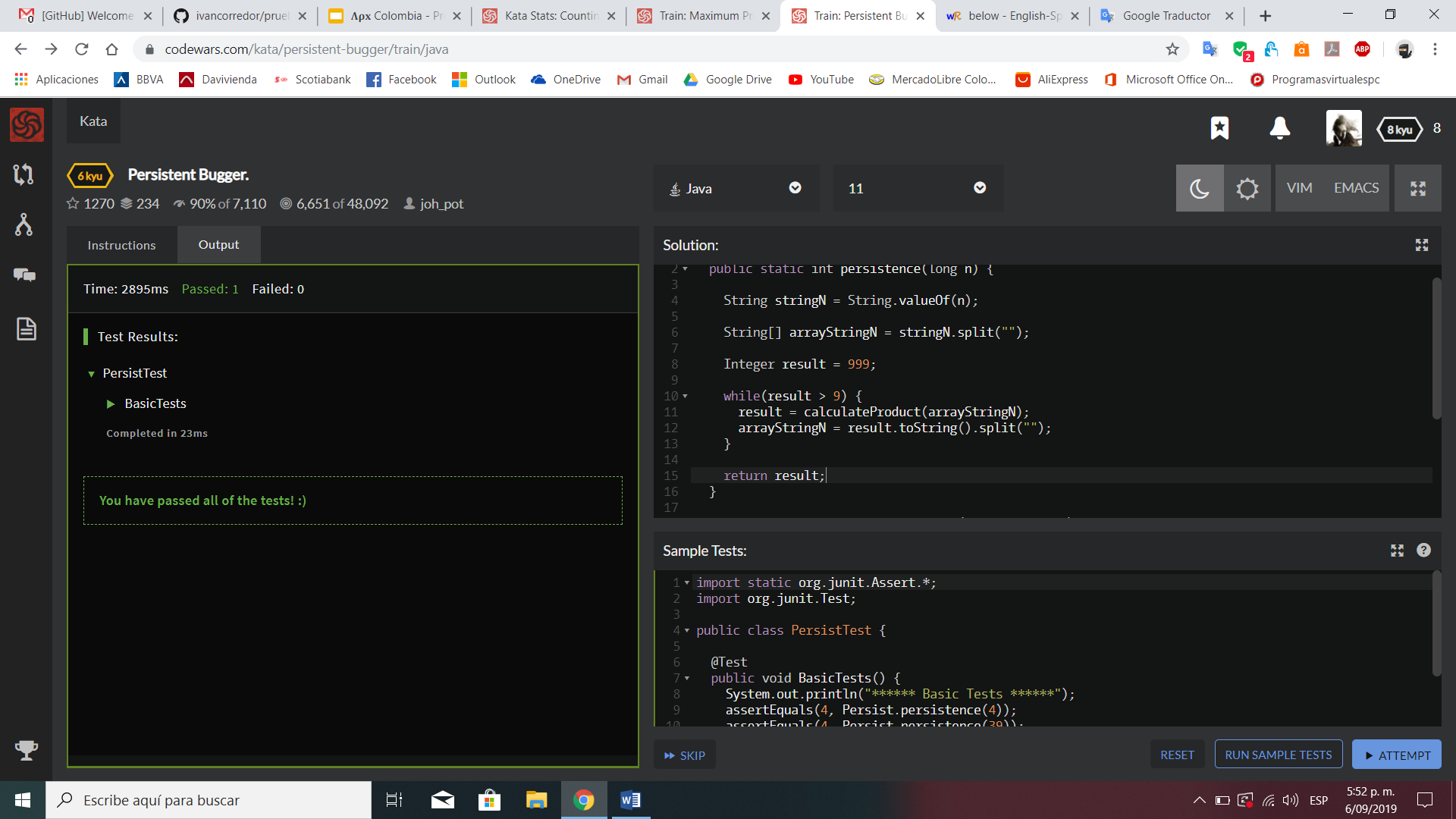
}

return maximumProduct;

}

}

Ejercicio 5



Código fuente:

class Persist {

public static int persistence(long n) {

String stringN = String.valueOf(n);

String[] arrayStringN = stringN.split("");

Integer result = 999;

while(result > 9) {

result = calculateProduct(arrayStringN);

arrayStringN = result.toString().split("");

}

return result;

}

public static int calculateProduct(String[] array) {

int result = 1;

for(int i = 0; i < array.length; i++) {

System.out.println("Numero en el array: " + array[i]);

result \*= Integer.parseInt(array[i]);

System.out.println("resultado: " + result);

}

return result;

}

}