Name: Sarwan Heer

CodeHS

Summative Coding Task

Main.java

```java
public class Main {
    public static void main(String[] args) {
        // Test shuffle method
        int[] first = {1, 3, 5, 7, 9};
        int[] second = {2, 4, 6};
        int[] shuffled = Shuffle.shuffle(first, second);
        System.out.print("Shuffled array: ");
        for (int num : shuffled) {
            System.out.print(num + " ");
        }
        System.out.println();

        // Test swap method
        int[][] array = {{1, 3, 5}, {2, 4, 6}};
        int[][] swapped = Swap.swap(array);
        System.out.println("Swapped array:");
        for (int[] row : swapped) {
            for (int num : row) {
                System.out.print(num + " ");
            }
            System.out.println();
        }

        // Test distinctChars method
        String word = "rhetorical";
        int distinctCount = distinctChars.distinctChars(word);
        System.out.println("Distinct characters in \"" + word + "\": " + distinctCount);

        // Test highestChars method (Bonus)
        String paragraph = "Consume a higher quantity of chicken, " +
        "the same you shall do for the pie. " +
        "Whatever does it matter whether the pastry was boiled or fried?" +
        "The question was that of the rhetorical variety!";

        String highestWord = highestChars.highestChars(paragraph);
        System.out.println("Word with highest distinct characters: " + highestWord);
    }
}
```

Shuffle.java

```java
public class Shuffle {
    public static int[] shuffle(int[] first, int[] second) {
        int minLength = Math.min(first.length, second.length);
        int[] result = new int[first.length + second.length];

        int index = 0;
        for (int i = 0; i < minLength; i++) {
            result[index++] = first[i];
            result[index++] = second[i];
        }

        if (first.length > minLength) {
            for (int i = minLength; i < first.length; i++) {
                result[index++] = first[i];
            }
        } else {
            for (int i = minLength; i < second.length; i++) {
                result[index++] = second[i];
            }
        }

        return result;
    }
}
```

Swap.java

```java
public class Swap {
    public static int[][] swap(int[][] array) {
        int rows = array.length;
        int cols = array[0].length;
        int[][] result = new int[cols][rows];

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                result[j][i] = array[i][j];
            }
        }

        return result;
    }
}
```

distinctChars.java

```java
import java.util.HashSet;

public class distinctChars {
    public static int distinctChars(String word) {
        HashSet<Character> uniqueChars = new HashSet<>();
        for (char c : word.toCharArray()) {
            uniqueChars.add(c);
        }
        return uniqueChars.size();
    }
}
```

highestChars.java

```java
import java.util.HashSet;

public class highestChars {
    public static String highestChars(String paragraph) {
        String[] words = paragraph.split("\\s+");
        String result = "";
        int maxDistinct = 0;

        for (String word : words) {
            HashSet<Character> uniqueChars = new HashSet<>();
            for (char c : word.toCharArray()) {
                uniqueChars.add(c);
            }
            if (uniqueChars.size() > maxDistinct) {
                maxDistinct = uniqueChars.size();
                result = word;
            }
        }
        return result;
    }
}
```