Gruppe: area51

Aufgabe 1 – 3 (oder siehe Blatt6.java)

```java
package Blatt6;

    public class Blatt6 {
        private static String rowResult = "";
        private static int amountOfRows = 0;
        private static boolean amountOfRowsIsSet = false;
        private static int currentIteration = 5;
        private static boolean invert = false;
        private static final StringBuilder stringBuilder = new StringBuilder();
        public static void main (String[] args) {
            int[] numbers1 = {};
            int[][] numbers2 = {{1,2,3},{1}};
            int[][][]numbers3 =  {{{1, 2}, {3}}, {{4}}};
            System.out.println(arraySum1D(numbers1));
            System.out.println(arraySum2D(numbers2));
            System.out.println(arraySum3D(numbers3));
            System.out.println(average3D(numbers3));
            System.out.println(symmetricDigitSequence(5));
            int[] array = { 2, 4, 1, 3, 7 };
            stepSum(array);
            stepSumRightAligned(array);
            symmetricDigitSequenceRec(5);
            System.out.println(stringBuilder);
        }

        public static int arraySum1D(int[] numbers) {
            int sum = 0;
            for (int i : numbers) {
                sum = sum + i;
            }
            return sum;
        }


        public static int arraySum2D(int[][] numbers) {
            int sum = 0;
            for (int[] i : numbers) {
                sum = sum + arraySum1D(i);
            }
            return sum;
        }
```

```
42
43      public static int arraySum3D(int[][][] numbers) {
44          int sum = 0;
45          for (int[][] i : numbers) {
46              sum = sum + arraySum2D(i);
47          }
48          return sum;
49      }
50
51      public static double average3D(int[][][] numbers) {
52          double avg = 0;
53          int length = 0;
54          for (int[][]i : numbers) {
55              length++;
56          }
57          if(length==0) {
58              return 0;
59          }else {
60              avg = (double)(arraySum3D(numbers)/length);
61              return avg;
62          }
63
64      }
65
66      public static String symmetricDigitSequence(int max) {
67          if (max < 0) {
68              return "Bitte positive Eingabe";
69          }
70          int value = max;
71          boolean reachedZero = false;
72          String result = "";
73          for (int i = 0; i<2*max+1 ; i++) {
74              result = result+value;
75              if(value>0 && !reachedZero) {
76                  value = value - 1;
77              }else {
78                  reachedZero = true;
79                  value = value + 1;
80              }
81          }
82          return result;
83      }
84
```

```java

    public static void stepSum(int[] arr) {
        if (arr.length==1) {
            System.out.println(arr[0]);
        }else {
            int newLength = arr.length;
            int[] newArr = new int[newLength-1];
            for (int i = 0; i<arr.length-1; i++) {
                newArr[i] = arr[i]+arr[i+1];
            }
            stepSum(newArr);
            for (int i : arr) {
                System.out.print(i+" ");
            }
            System.out.println();
        }
    }
    public static void stepSumRightAligned(int[] arr) {
        if (!amountOfRowsIsSet) {
            amountOfRows = arr.length * 2;
            amountOfRowsIsSet = true;
        }
        if (arr.length == 1) {
            System.out.printf("%10s", arr[0]);
            System.out.println();

        } else {
            int newLength = arr.length;
            int[] newArr = new int[newLength-1];
            for (int i = 0; i < arr.length - 1; i++) {
                newArr[i] = arr[i]+arr[i+1];
            }
            stepSumRightAligned(newArr);
            for (int i : arr) {
                rowResult += i + " ";
            }
            System.out.printf("%11s", rowResult);
            amountOfRows--;
            rowResult = " ";
            System.out.println();
        }
    }
```

```java
    public static void symmetricDigitSequenceRec(int max) {
        if (currentIteration == 0 && !invert) {
            invert = true;
        }
        stringBuilder.append(currentIteration);

        // Counter
        if (invert) {
            currentIteration++;
        } else {
            currentIteration--;
        }

        // Exit condition
        if (currentIteration > max) {
            return;
        }

        // Executions
        symmetricDigitSequenceRec(max);
    }
}
```

Aufgabe 4 (oder siehe FourToAnyNumber.java):

```java
package Blatt6;

import java.util.Scanner;

public class FourToAnyNumber {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Zahl (> 0): ");
        int input = scanner.nextInt();
        solve(input);
        scanner.close();
    }

    public static void solve(int val) {
        if (val<0) {
            System.out.println("Eingabe muss größer als 0 sein.");
        }else if (val == 4) {
            System.out.println("4");
        }else {
            String input = ""+val;
            String result = "";
            int length = input.length();
            char lastDigit = input.charAt(length-1);
            char[] chars = input.toCharArray();
            int val2 = 0;
            if (lastDigit == '0' || lastDigit== '4') {
                for (int i = 0 ; i<length-1;i++) {
                    result = result + chars[i];
                    val2 = Integer.parseInt(result);
                }
            }else {
                val2 = val * 2;
            }
            solve(val2);
            System.out.println(val);
        }

    }

}
```