

5/5

Aufgabe 1

```
package uebung06;
```

```
public class Uebung061 {
```

```
    public static void main(String[] args) {
```

```
        // TODO Auto-generated method stub
```

```
    }
```

```
    public static int arraysum1D(int[] numbers) {
```

```
        int result = 0;
```

```
        for (int i = 0; i < numbers.length; i++)
```

```
        {
```

```
            result += numbers[i];
```

```
        }
```

```
        return result;
```

```
}
```

```
public static int arraysum2D(int[][] numbers) {  
    int result = 0;  
  
    for(int index = numbers.length - 1; index >= 0; index--) {  
        result = result + arraysum1D(numbers[index]);  
    }  
  
    return result;  
}
```

```
public static int arraysum3D(int[][][] numbers) {  
    int result = 0;  
  
    for(int index = numbers.length - 1; index >= 0; index--) {  
        result = result + arraysum2D(numbers[index]);  
    }  
}
```

```
return result;
```

```
}
```

```
public static double average3D(int[][][] numbers) {
```

```
    if(numbers.length != 0) {
```

```
        double result = 0;
```

```
        double n = 0;
```

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

```
            for(int j = 0; j < numbers[i].length; j++) {
```

```
                for(int k = 0; k < numbers[i][j].length; k++) {
```

```
                    n = n + 1;
```

```
                }
```

```
            }
```

```
        }
```

```
        result = arraysum3D(numbers) / n;
```

```
        return result;
```

```
    }
```

```
else {  
    return 0;  
}
```

```
}
```

```
}
```

4/4

Aufgabe 2

```
package uebung06;
```

```
public class uebung062 {
```

```
    public static void main(String[] args) {
```

```
        // TODO Auto-generated method stub
```

```
        System.out.println(symmetricDigitSequence(4));
```

```
        System.out.println(symmetricDigitSequenceRed(4));
```

```
    }
```

```
    public static String symmetricDigitSequence(int max) {
```

```
        //positive Zahlen das sollte eigentlich auch nur für positive Zahlen funktionieren, also die Implementierung für negative Zahlen wäre nicht nötig gewesen
```

```
        if(max >= 0) {
```

```
            int count = max;
```

```
            String a = "";
```

```
            for( int i = count; i > 0; i-- ) {
```

```
        a = a + i;  
    }
```

```
String b = "0";
```

```
String c = "";
```

```
for( int j = 1; j <= count; j++ ) {  
    c = c + j;  
}
```

```
String result = a + b + c;
```

```
return result;  
}
```

```
//negative Zahlen
```

```
else {  
    int count = max;  
    String a = "";  
    for( int i = count; i < 0; i++ ) {
```

```
        a = a + i;
    }

    String b = "0";

    String c = "";

    for( int j = -1; j >= count; j-- ) {
        c = c + j;
    }

    String result = a + b + c;

    return result;
}
}
```

```
public static String symmetricDigitSequenceRec(int max){
    //positive Zahlen
    if (max >=0) {
```

```
if( max == 0) {  
    return "0";  
}
```

```
else {  
    return max + symmetricDigitSequenceRec(max - 1) + max;
```

```
}
```

```
}
```

```
//negative Zahlen
```

```
else {  
    if( max == 0) {  
        return "0";  
    }  

```

```
else {  
    return max + symmetricDigitSequenceRec(max + 1) + max;
```


}

}

}

}

2/8

Aufgabe 3

package uebung06; Die Ausgabe sollte andersherum erfolgen (das Eingabearray als unterstes)

public class Uebung063 {

```
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        int[] array1 = new int [] {1, 3, 4, 5, 6};  
        stepSum(array1);  
    }
```

```
    public static int[] stepSum(int[] array) {  
        for(int values: array) {  
            System.out.print(values + " ");  
        }  
        System.out.println("");  
  
        if(array.length == 0) {
```

```
int[] a = new int[] {};  
return a;  
}
```

```
else {  
int[] array2 = new int[array.length - 1];  
for(int i = 0; i < array.length - 1; i++) {  
  
    array2[i] = array[i] + array[i + 1];  
}  
  
stepSum(array2);  
return array2;  
}
```

```
}
```

```
}
```


2,5/3

Aufgabe 4:

```
package uebung06;
```

```
import java.util.Scanner;
```

```
public class Uebung064 {
```

die Ausgabe müsste anderherum sein (ausgehend von der 4 zur eingegeben Zahl)

```
    public static void main(String[] args) {
```

```
        // TODO Auto-generated method stub
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        System.out.print("Zahl (> 0): ");
```

```
int zahl = scanner.nextInt();  
scanner.close();  
System.out.println(vier(zahl));  
  
}
```

```
public static int vier(int zahl) {  
    //prüfen ob zahl > 0  
    if(zahl > 0) {  
        // Basisfall  
        if(zahl == 4) {  
            return 4;  
        }  
    }  
}
```

```
//letzte Zahl = 4 entfernen
```

```
else if((zahl - 4) % 10 == 0) {  
    System.out.println(zahl);  
    return vier((zahl - 4) / 10);  
}
```

Diesen und den übernächsten Fall hätte man in einem Block zusammenfassen können

```
}
```

```
//letzte Zahl = 2 --> verdoppeln
```

```
else if((zahl - 2) % 10 == 0) {
```

dieser Fall wird auch durch das letzte else abgedeckt

```
    System.out.println(zahl);
```

```
    return vier(zahl * 2);
```

```
}
```

```
else if(zahl % 10 == 0) {
```

```
    System.out.println(zahl);
```

```
    return vier(zahl / 10);
```

```
}
```

```
else {
```

```
    System.out.println(zahl);
```

```
    return vier(zahl * 2);
```

```
}
```

```
}
```

```
else {
```

```
return -00000;
```

```
}
```

```
}
```

```
}
```

13,5/20