

# Fachhochschule Bielefeld

## Campus Minden

### Objektorientierte Programmierung

Prof. J.Rexilius

Wael Eskeif, 16.11.2022

Inhalt:

### 3. Aufgabe

(a). Memory .....	-2-
(b). Einkaufswagen.....	-4-
(c). Bild vergrößern .....	-6-

## (a). Memory

- Zuerst müsste zwei Klassen erstellt werden, die erste als Memory1, wo die Methoden erstellt werden müssten, und die Zweite als MemoryTest1, wo die Main Methode ist und den gesamten Code abgespielt wird.
- Als nächste Schritt sollte ein zweidimensionales Array als Speicherplatz erstellt werden und als Private definieren,
- Danach musste ein zweidimensionales Array Methode mit einem Parameter n erstellt werden, und die Länge des Arrays wird mit dem Parameter n festgestellt
- Mithilfe einer Formel  $n \times n / 2$  wird Anzahl der Paar-Zahlen festgestellt und in Size als Int gespeichert.
- Und danach wird der Parameter n geprüft, ob null oder eine ungerade Zahl ist, wenn ja, dann sollte nichts gemacht werden, wenn nicht, dann werden Zwei For-Schleifen verwendet und danach wird der Paarzähler gegen null gesetzt werden, und als nächste Schritt wird eine Fallunterscheidung benutzt, damit geprüft wird, ob der Counter kleiner als Size ist, wenn der Fall ist dann werden zwei randomZahlen erstellt einmal für Zeilenrandom und einmal für die Spaltenrandom.
- Und jetzt wird mit If-Bedingung geprüft, wo leer ist, wird die Anfangszahl 5 eingesetzt und nach jeden Durchlauf um eins erhöht und somit ist die Funktion bereit.
- Noch eine weitere Methode für die Ausgabe ist auch relevant zu machen, durch zwei For-Schleifen werden alle Elemente ausgedruckt.
- Das Ganze wird in MemoryTest1 Klasse abgespielt.

```
import java.util.Scanner;

public class MemoryTest1 {

    public static void main(String[] args) {
        // Ein Objekt erstellt.
        Memory1 memory = new Memory1();
        // Die Methoden aufrufen.
        memory.generateField(4);
        memory.ausgabe();
        System.out.println("-----");
        memory.generateField(6);
        memory.ausgabe();
    }
}
```

Run: MemoryTest1

```
10 8 6 6
5 12 9 7
11 9 7 5
10 12 11 8
-----
```

Build completed successfully in 2 sec, 516 ms (moments ago)

## Test2

```
import java.util.Scanner;

public class MemoryTest1 {

    public static void main(String[] args) {
        // Ein Objekt erstellt.
        Memory1 memory = new Memory1();
        // Die Methoden aufrufen.
        memory.generateField(4);
        memory.ausgabe();
        System.out.println("-----");
        memory.generateField(6);
        memory.ausgabe();
    }
}
```

Run: MemoryTest1

```
13 20 21 6 7 5
14 13 8 22 8 17
9 15 16 7 22 12
15 10 18 10 17 14
19 18 11 5 11 9
19 12 21 6 20 16
```

Process finished with exit code 0

Build completed successfully in 2 sec, 516 ms (a minute ago)

Wael Eskeif, 16.11.2022

-3-

(b). Einkaufswagen.

- Zuerst müsste zwei Klassen erstellt werden, die erste als Einkaufswagen, wo die Methoden erstellt werden müssten, und die Zweite als Einkaufswagen Test, wo die Main Methode ist und den gesamten Code abgespielt wird.
- Dann wird die ArrayList Klasse importiert und ein Objekt davon erzeugen und als Einkauf gespeichert.
- Als nächste Schritt müssten vier Methoden insgesamt erstellt werden, die erste Methode als addToCart () mit dem Parameter item als String erstellt, und darein wird überprüft, ob Länge des Arrays kleiner als 5 ist, wenn ja, dann wird hinzugefügt.
- Die zweite Methode removeFromCart() mit dem Parameter item, da wird das Objekt gelöscht.
- Die dritte Methode showContent() sollte erstellt werden, darein sollte eine For-Schleife verwendet werden, um alle Elemente auszudrucken.
- Die vierte und die letzte Methode Size(), da wird nur die Länge des Arrays ausgedruckt.

## Test

The screenshot shows an IDE with a project named 'Memory'. The project structure includes a 'src' directory with files 'Einkaufswagen.java', 'EinkaufswagenTest.java', 'Main.java', 'Memory.java', and 'MemoryTest.java'. The 'EinkaufswagenTest.java' file is open, showing the following code:

```

1 public class EinkaufswagenTest {
2     public static void main(String[] args) {
3         // Cart Objekt von der Klasse Einkaufswagen.
4         Einkaufswagen cart = new Einkaufswagen();
5         cart.addToCart( item: "Birne");
6         cart.addToCart( item: "Salat");
7         cart.addToCart( item: "Paprika");
8         cart.showContent();
9         cart.removeFromCart( item: "Salat");
10        System.out.println();
11        cart.showContent();
12        cart.size();
13    }
14 }

```

The 'Run' console at the bottom shows the output of the program:

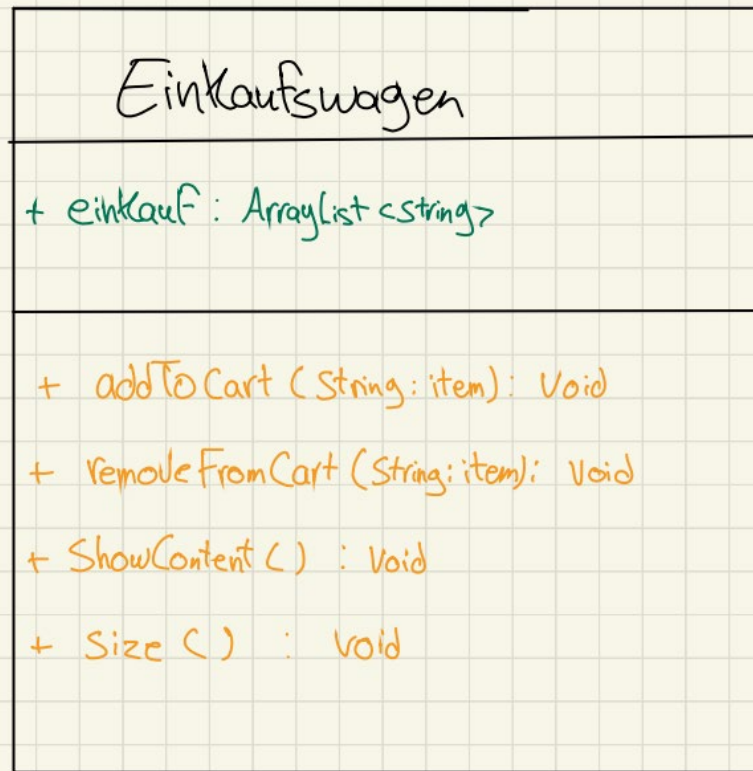
```

C:\Users\weskeif\.jdk\corretto-18.0.2\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2022.2.3\lib\idea_rt.jar=52653:C:\Program
Birne
Salat
Paprika

Birne
Paprika
2
Process finished with exit code 0

```

## Klassendiagramm.



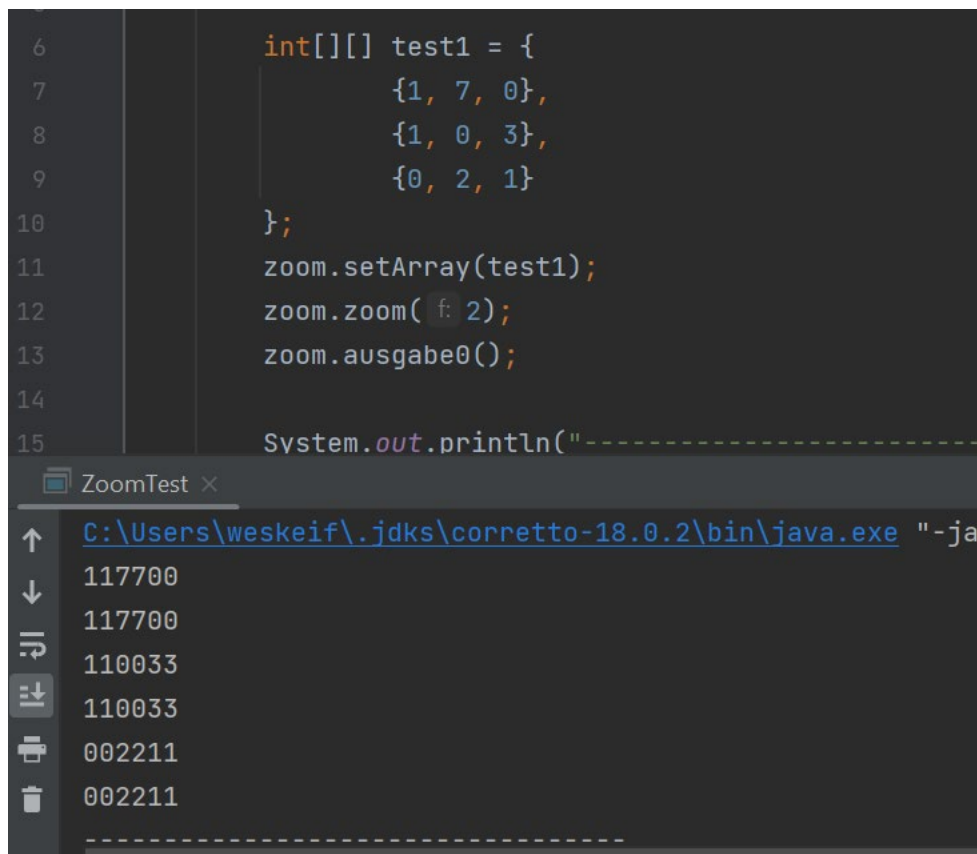
- Zuerst müsste zwei Klassen erstellt werden, einmal Klasse Zoom, wo der Code und die Methoden geschrieben werden, und Klasse zoomtest, wo die Main Methode verwendet und implementiert wird.
- Als Erstes wird das Array geprüft werden, ob es null ist, und als nächste Schritt wird die Graustufen überprüft, ob das Eingebens Array eine Zahl größer als 7 hat, falls ja, wird das Programm beendet.
- Und jetzt wird die den neuen Array mit der Abhängigkeit mit dem Parameter festgestellt, danach wird jeden Einzel Element von dem alten Array in den bestimmten Positionen im neuen Array geschrieben.
- In der Test-Klasse werden wird das eingegebenes Array eingegeben und wird durch den Aufruf von den Methoden in Zoom klasse bearbeitet und ausgegeben.

Test1,F "2"

```

6      int[][] test1 = {
7          {1, 7, 0},
8          {1, 0, 3},
9          {0, 2, 1}
10     };
11     zoom.setArray(test1);
12     zoom.zoom(f: 2);
13     zoom.ausgabe0();
14
15     System.out.println("-----

```



The screenshot shows a Java IDE with a code editor and a console window. The code editor contains a Java program that initializes a 3x3 integer array 'test1' with values [[1, 7, 0], [1, 0, 3], [0, 2, 1]]. It then creates a 'zoom' object, sets the array, calls 'zoom(2)', and calls 'ausgabe0()'. The console window shows the output of the program, which is a 3x3 grid of memory addresses: 117700, 117700, 110033 in the first row; 110033, 002211, 002211 in the second row; and 002211, 002211, 002211 in the third row. The console window title is 'ZoomTest x'.

Test1,F "3"

```
5
6      int[][] test1 = {
7          {1, 7, 0},
8          {1, 0, 3},
9          {0, 2, 1}
10     };
11     zoom.setArray(test1);
12     zoom.zoom(f: 3);
```

Run: ZoomTest x

C:\Users\weskeif\.jdk\corretto-18.0.2\bin\java.exe "-javaagent:C:\Program Fi

111777000  
111777000  
111777000  
111000333  
111000333  
111000333  
000222111  
000222111  
000222111



Test2,F "2"

```
15      System.out.println("-----");
16      int[][] test2 = {
17          {1, 0, 2},
18          {4, 2, 4},
19          {2, 0, 1}
20      };
21      zoom.setArray(test2);
22      zoom.zoom(f: 2);
```

ZoomTest x

↑ -----  
↓ 110022  
↕ 110022  
↕ 442244  
↕ 442244  
🖨 220011  
🗑 220011

Test2,F "3"

```
int[][] test2 = {
    {1, 0, 2},
    {4, 2, 4},
    {2, 0, 1}
};
zoom.setArray(test2);
zoom.zoom(f: 3);
zoom.ausgabe0();
```

ZoomTest x

-----

111000222  
111000222  
111000222  
444222444  
444222444  
444222444  
222000111  
222000111  
222000111

Process finished with exit code 0