

Aufgabe 1: Arrays

Lösen Sie die Aufgaben 4.46 bis 4.51 im Lehrbuch.

4.46:

```
readings = new double[60];
urls = new String[90];
machines = new TicketMachine[5];
```

4.47:

20 Stück (von 0 bis 19)

4.48:

[] anstelle von ()

```
double[] prices = new double[50];
```

4.49:

ArrayIndexOutOfBoundsException

4.50:

```
public void printHourlyCounts()
{
    int hour = 0;
    System.out.println("Hr: Count");
    while (hour < hourCounts.length) {
        System.out.println(hour + ": " + hourCounts[hour]);
        hour++;
    }
}
```

4.51:

```
public void printGreater(double marks, double mean)
{
    for(index = 0; index <= marks.length; index++) {
        if(marks[index] > mean) {
            System.out.println(marks[index]);
        }
    }
}
```

```
public void printGreater(double[] marks, double mean)
{
    for (int index = 0; index < marks.length; index++) {
        if (marks[index] > mean) {
            System.out.println(marks[index]);
        }
    }
}
```

Aufgabe 2: Arrays, Methoden und Debugger

Übernehmen Sie folgendes Gerüst für eine Klasse Aufgabe2:

```
public class Aufgabe2
{
    private int[] a;
    public Aufgabe2(int length)
    {
        a = new int[length];
    }
    ...
}
```

Implementieren Sie folgende Methoden:

a) `public void setFix(int value)`: Initialisiert alle Felder von `a` mit dem Wert von `value`.

```
public void setFix(int value)
{
    for (int i=0; i < a.length; i++) {
        a[i] = value;
    }
}
```

b) `public void setup()`: Initialisiert `a` aufsteigend mit Werten 1, 2, 3, ...

```
public void setup()
{
    for (int i=0; i < a.length; i++) {
        a[i] = i+1;
    }
}
```

c) `public void setRandom(int min, int max)`: Initialisiert `a` zufällig mit Werten aus dem Bereich `[min, max]` (vgl. mit dem Beispiel aus der Vorlesung).

```
public void setRandom(int min, int max)
{
    for (int i=0; i < a.length; i++) {
        a[i] = (int)(Math.random()*max) + min;
    }
}
```

d) `public void add(int value)`: Addiert zu allen Feldern von `a` den Wert `value`.

```
public void add(int value)
{
    for (int i=0; i < a.length; i++) {
        a[i] += value;
    }
}
```

e) `public boolean add(int[] b)`: Addiert `b` elementweise zu `a` (falls überhaupt möglich!). Bei erfolgreicher Addition ist der Rückgabewert `true`.

```
public boolean add(int[] b)
{
    boolean res = false;

    if (a.length >= b.length) {
        for (int i=0; i < b.length; i++) {
            a[i] += b[i];
        }
        res = true;
    }

    return res;
}
```

Überprüfen Sie das Funktionieren Ihrer Methoden mit Hilfe des Object -Inspectors und Debuggers von BlueJ.

Hinweis: Ein Array lässt sich in BlueJ via Eingabe-Fenster direkt wie folgt als Parameter einer Methode übergeben, z.B.: `{1, 4, 7, -10}`

done.

Aufgabe 3: Gleichwertigkeit der Schleifen-Anweisungen

Lösen Sie die Aufgabe 4.50 im Lehrbuch (`while` → `for`).

4.50:

Siehe Aufgabe 1.

Aufgabe 4: Klasse GuestBook

Implementieren Sie eine einfache Klasse GuestBook. Ein Gästebuch kann beliebig viele Einträge (vgl. Zeichenketten in einer ArrayList) aufnehmen. Die Anzahl Einträge können abgefragt und das Gästebuch kann auf die Konsole zeilenweise ausgegeben werden.

Entsprechend stehen folgende Methoden im Zentrum:

a) public void insertEntry(String note)

b) public int getNoOfEntries()

c) public void printEntries()

```
import java.util.ArrayList;

/**
 * GuestBook
 *
 * @author Felix Rohrer
 * @version 27.10.2011 v0
 */
public class GuestBook
{
    // instance variables - replace the example below with your own
    private ArrayList<String> myGuestBook;

    /**
     * Constructor for objects of class GuestBook
     */
    public GuestBook()
    {
        // initialise instance variables
        myGuestBook = new ArrayList<String>();
    }

    /**
     * Add a new GuestBook entry
     *
     * @param note new guestbook entry
     */
    public void insertEntry(String note)
    {
        myGuestBook.add(note);
    }

    /**
     * Get the number of entries
     *
     * @return number of entries
     */
    public int getNoofEntries()
    {
        return myGuestBook.size();
    }

    /**
     * Print Guestbook
     */
    public void printEntries()
    {
        for(String note : myGuestBook) {
            System.out.println(note);
        }
    }
}
```