

Übung 6

Zufallszahlen, Arrays, Strings und Funktionen

Zufallszahlen

In diesem Aufgabenblatt müssen an verschiedenen Stellen Zufallszahlen erzeugt werden. Der folgende Programmausschnitt zeigt, wie Zufallszahlen mit Hilfe von **double Math.random()** erzeugt werden können.

```
public static void main(String[] args) {  
  
    // Zufallszahl zwischen 0 und 1  
    System.out.println(Math.random());  
  
    // Zufallszahl zwischen 1 und 10  
    int zufall10 = (int) (Math.random()*10)+1;  
  
}
```

Aufgabe 1: Was gibt das Programm aus? (1 Punkt)

Gegeben sei das folgende Java-Programm:

```
public static void main(String[] args) {  
    int [] a = {1,2,3};  
    int [] b = {4,5,6};  
    int [] c = b = a;  
    b[1] = a[2];  
    c[2]++;  
    a[0]--;  
  
    System.out.println(a[0] + " " + a[1] + " " + a[2]);  
    System.out.println(b[0] + " " + b[1] + " " + b[2]);  
    System.out.println(c[0] + " " + c[1] + " " + c[2]);  
}
```

Was gibt das Programm aus? Erklären Sie, was passiert.

Aufgabe 2: Arrays (2 Punkte)

- Deklarieren Sie ein `int`-Array mit 100 Elementen und füllen Sie es mit den ersten 100 Zahlen absteigend sortiert (100, 99, 98, ...).
- Deklarieren Sie ein `int`-Array mit 100 Elementen und füllen Sie es mit den ersten 100 *Quadratzahlen*.

Aufgabe 3: Lotto (4 Punkte)

Spielen Sie (virtuell) Lotto! Schreiben Sie dazu ein Java-Programm, welches zunächst 6 Zahlen (1 - 49) von der Konsole einliest und diese in einem Array abspeichert. Das Einlesen der Zahlen soll in einer eigenen Methode **tipp** passieren. Jede Zahl darf aber nur genau 1 mal getippt werden (keine Zahl darf doppelt vorkommen)!

```
public static int [] tipp() { ... }
```

Wurden die getippten Zahlen eingelesen findet eine Ziehung statt in welcher 6 Zufallszahlen zwischen 1 und 49 ermittelt werden. Schreiben Sie hierzu eine Methode **ziehung** welche 6 Zufallszahlen ermittelt, in einem Array abspeichert und anschließend auf dem Bildschirm ausgibt. Auch hier darf eine Zahl nicht doppelt gezogen werden!

```
public static int [] ziehung() { ... }
```

Implementieren Sie nun eine Funktion **auswertung** welche bestimmt wie viele **Richtige** der Spieler getippt hat und geben Sie das als Ergebnis auf dem Bildschirm aus.

```
public static int auswertung(int [] tipp, int [] ziehung)
```

Aufgabe 4: Sätze (3 Punkte)

In dieser Aufgabe sollen zufällige Sätze gebildet werden. Hierzu soll ein mehrdimensionales Array, welches die folgende Tabelle repräsentiert, angelegt und mit Worten wie im Beispiel gefüllt werden. Bilden Sie nun per Zufall 10 Sätze und geben Sie diese auf dem Bildschirm aus.

1. Wort	2. Wort	3. Wort
Der Hausmeister	bäckt	gerne
Die Katze	tanz	verliebt
Der Student	singt	niemals
Der Bäcker	arbeitet	entspannt
Die Programmiererin	fährt	schnell