



## Erzeugen von Zufallszahlen

**Aufgabe:** Ein Array soll mit Zufallszahlen gefüllt und dann angezeigt werden.

### Balkendiagramm

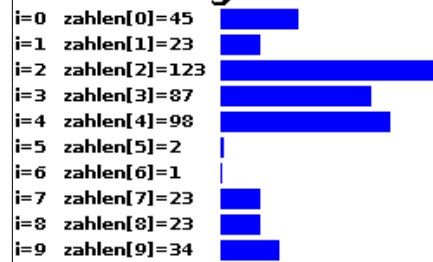


Bild: Screenshot von Ausführung des Programms „balkendiagramm“ (Schaller)

## Umgang mit Arrays (Wiederholung):

### 1. Schritt: Deklaration

Um ein Array eines Variablentyps zu deklarieren, muss an den Typ eine eckige Klammer angehängt werden.

z.B. `double[] kommazahlen;`  
`char[] buchstaben;`

### 2. Schritt: Initialisierung

Es muss genügend Platz für alle Einträge im Speicher reserviert werden (Befehl `new`). Daher muss man jetzt die Größe des Arrays festlegen.

z.B. `kommazahlen = new double[20];`  
`buchstaben = new char[100];`

### 3. Schritt: Werte der Einträge festlegen.

In der Regel werden in einer `for`-Schleife alle Werte des Arrays einzeln belegt.

z.B. 

```
for(int i=0; i < kommazahlen.length; i++) {
    kommazahlen[i] = 0.0;
}
```

## Aufgabe Implementation:

Lade das Programm `alg11_balkendiagramm_zufall`. Du musst dort die Methode `erzeugeZufallsarray(int laenge)` implementieren.

1. Initialisiere das globale Array `zahlen` in der Methode `erzeugeZufallsarray()`, so dass es 4 ganze Zahlen speichern kann. Lege die Werte der Einträge auf 20, 50, 30 und 100 fest.  
 Teste, ob das Array korrekt angezeigt wird.
2. Fülle die vier Arrayelemente mit Zufallszahlen zwischen 1 und 300. Verwende dazu die Methode `getZufallszahl(int min, int max)`, die Zufallszahlen zwischen `min` und `max` erzeugt.
3. Verwende den Parameter `laenge`, um die Größe des Arrays festzulegen. Fülle in einer `for`-Schleife das ganze Array mit Zufallszahlen zwischen 1 und 300.

## Testen der Implementation:

4. Überlege dir mögliche Grenzfälle. Bei welchen Werten von `laenge` könnte es Probleme geben? Bei welchen Zufallszahlen könnte es schwierig werden?
5. Teste deine Grenzfälle und passe ggf. das Programm daran an. Teste z.B. mit einer `if`-Anweisung, ob ein zulässiger Wert für `laenge` übergeben wurde.