

# ArrayConcat

## Frage 1 [12 Punkte]

Definieren Sie eine (statische) Methode "concat", die zwei int-Arrays als Eingabe erhält und ein neues (drittes) int-Array als Ergebnis liefert. Das Ergebnis-Array soll dabei aus dem "Aneinanderhängen" ("concat") der beiden Eingabe-Arrays entstehen.

In [0]: `### BEGIN SOLUTION`

```
public static int[] concat(int[] a, int[] b) {
    int[] erg = new int[a.length + b.length];
    for (int i = 0; i < a.length; i++)
        erg[i] = a[i];
    for (int j = 0; j < b.length; j++)
        erg[a.length + j] = b[j];
    return erg;
}
```

`### END SOLUTION`

## TraceTabelle

## Frage 2 [8 Punkte]

Erstellen Sie eine Trace-Tabelle für die folgende Methode, wenn sie für den Wert n=10 aufgerufen wird.

```
01: public static int test(int n) {
02:     int sum = 0;
03:     int z = 1;
04:     int anz = 0;
05:     while(sum < n) {
06:         sum = sum + z;
07:         z = z + 2;
08:         anz++;
09:     }
10:     return anz;
11: }
```

Benutzen Sie folgendes Format für die Tabelle:

Zeile	n	sum	z	anz	sum<n
-------	---	-----	---	-----	-------

(Füllen Sie bitte JEDE Tabellenposition aus und trennen Sie die Spalten mit einem senkrechten Strich "|". Lassen Sie keine Tabellenpositionen frei, da wir dann möglicherweise die Einträge nicht mehr korrekt den Spalten zuordnen können. Sie können "leere" Tabellenpositionen mit einem Punkt "." füllen. Zur Platz- und Zeitersparnis

dürfen Sie die Auswirkung mehrerer Codezeilen in einer Tabellenzeile zusammenfassen.)

In [0]:

[illegible]

#### BEGIN SOLUTION

### # Answer

Zeile	n	sum	z	anz	sum < n
1	10	.	.	.	.
2-4	10	0	1	0	.
5	10	0	1	0	true
6-8	10	1	3	1	.
5	10	1	3	1	true
6-8	10	4	5	2	.
5	10	4	5	2	true
6-8	10	9	7	3	.
5	10	9	7	3	true
6-8	10	16	9	4	.
5	10	16	9	4	false
10	10	16	9	4	.

```
### END SOLUTION
```