

HOPP ITs



Kartenbeschreibung

KARTENBESCHREIBUNG

| OPERATOREN | BEDEUTUNG |
|-------------------------------|---|
| <code>zahl = zahl + 4;</code> | Erhöht den Wert der Variablen »zahl« um 4. |
| <code>zahl = zahl - 2;</code> | Verringert den Wert der Variablen »zahl« um 2. |
| <code>zahl = zahl * 3;</code> | Multipliziert den Wert der Variablen »zahl« mit 3. |
| <code>zahl++;</code> | Erhöht den Wert der Variablen »zahl« um 1. |
| <code>zahl--;</code> | Verringert den Wert der Variablen »zahl« um 1. |
| <code>zahl%2;</code> | Rest beim Teilen durch 2, z. B. $7/2 = 3$ Rest 1 , also ist $7\%2 = 1$. |
| wenn zahl negativ: | $-8/5 = -1$ Rest -3 . Da Computer immer positive Zahlen beim Modulo-Operator ausgeben, ist $-8\%5 = 2$ ($2 = -3 + 5$). |
| ZUWEISUNG | BEDEUTUNG |
| <code>zahl = 7;</code> | Weist der Variablen »zahl« den Wert 7 zu. |

| VERGLEICHS-OPERATOREN | BEDEUTUNG |
|--|--|
| <code>zahl == 5;</code> | Prüft, ob die Variable »zahl« den Wert 5 hat. Gibt einen booleschen Wert zurück (wahr/falsch). |
| <code>zahl > 3; zahl <= 3;</code> | Prüft, ob der Wert der Variablen »zahl« größer 3 oder kleiner gleich 3 ist. Gibt einen booleschen Wert zurück (wahr/falsch). |
| BEDINGTE ANWEISUNGEN | BEDEUTUNG |
| <code>if (zahl == 6) { Anweisung; }</code> | Führt die Anweisung nur dann aus, wenn die Bedingung »zahl==6« erfüllt (wahr) ist. |
| <code>if ((zahl > 0) && (zahl < 20)) { Anweisung; }</code> | Führt die Anweisung nur dann aus, wenn beide Bedingungen erfüllt sind, also: »zahl > 0« und »zahl < 20«. |

SCHLEIFEN

BEDEUTUNG

```
while (Bedingung) {  
Anweisung;  
}
```

Führt die Anweisung aus, solange die Bedingung wahr ist.

```
for (ANFANGSANWEISUNG;  
BEDINGUNG; ENDANWEISUNG) {  
Anweisung;  
}
```

```
for (int i=0; i<=3; i++) {  
Anweisung;  
}
```

Dies ist eine For-Schleife.

- › Bei der **Anfangsanweisung** wird die Variable »i« auf den Wert 0 gesetzt.
- › Solange die **Bedingung** erfüllt ist, wird die Anweisung ausgeführt. Wenn sie nicht erfüllt ist, wird die Schleife beendet.
- › Nach jedem Schleifendurchlauf wird die **Endanweisung** ausgeführt. Hier wird der Wert der Variablen »i« um 1 erhöht.

Im obigen Beispiel wird die Anweisung genau 4-mal ausgeführt, nämlich für die Werte $i=0$, $i=1$, $i=2$ und $i=3$.