

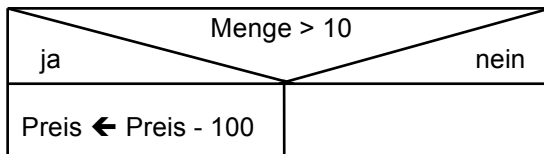
Struktogramm	Wortform	C-Code
<p><b>1. Anweisungsblock</b></p> <pre>       hilf ← a       a   ← b       b   ← hilf           </pre>	<p>tausche a, b</p>	<pre> {   hilf = a;   a   = b;   b   = hilf; }           </pre>
<p><b>2. Bedingte Anweisung: Verzweigung einseitig</b></p> <pre>       /----- x ≠ 0 -----\      /                         \     ja                         nein                                           d ← 0                              y ← 1/x                                                              -----\           </pre>	<p>wenn ( <math>x \neq 0</math> )  <math>d = 0</math>  <math>y = 1/x</math></p>	<pre> if (x != 0) {   d = 0;   y = 1/x; }           </pre>
<p><b>zweiseitig</b></p> <pre>       /----- x ≠ 0 -----\      /                         \     ja                         nein                                           d ← 1                              y ← 1/x                           -----\       -----\        d ← 0                              y ← 0                             -----\           </pre>	<p>wenn ( <math>x \neq 0</math> )  <math>d = 1</math>  <math>y = 1/x</math>    sonst  <math>d = 0</math>  <math>y = 0</math></p>	<pre> if (x != 0) {   d = 1;   y = 1/x; } else {   d = 0;   y = 0; }           </pre>
<p><b>mehrseitig</b></p> <pre>       /----- falls s = -----\      /-----\     1         2         3         sonst                                             a ← 5     a ← 3     a ← 1     a ← 0       ----- ----- -----            </pre>	<p>Wahlschalter (s)  falls 1: <math>a = 5</math>  falls 2: <math>a = 3</math>  falls 3: <math>a = 1</math>  sonst : <math>a = 0</math></p>	<pre> switch (s) {   case 1 : a = 5; break;   case 2 : a = 3; break;   case 3 : a = 1; break;   default: a = 0; }           </pre>
<p><b>3. Wiederholung: Schleife mit Zähler</b></p> <pre> wiederhole für z ← 1 bis 10   s ← s + z           </pre>	<p>für z = 1 bis 10 Schritt 1 führe  <math>s = s + z</math>  aus</p>	<pre> for (z = 1; z &lt;= 10; z++) {   s = s + z; }           </pre>
<p><b>kopfgesteuert: Anfangsbedingung</b></p> <pre> wiederhole solange s &lt; 100   s ← s + z   z ← z + 1           </pre>	<p>solange <math>s &lt; 100</math> führe  <math>s = s + z</math>  <math>z = z + 1</math>  aus</p>	<pre> while (s &lt; 100) {   s = s + z;   z = z + 1; }           </pre>
<p><b>fußgesteuert: Endebedingung</b></p> <pre>   s ← s + z   z ← z + 1 wiederhole solange s &lt; 100           </pre>	<p>wiederhole  <math>s = s + z</math>  <math>z = z + 1</math>  solange <math>s &lt; 100</math></p>	<pre> do {   s = s + z;   z = z + 1; } while (s &lt; 100);           </pre>

## Vergleiche

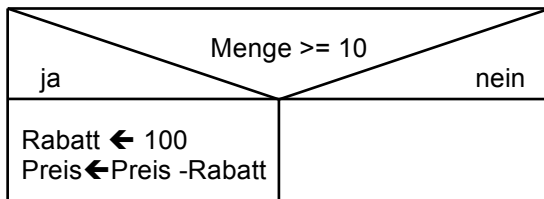
: > >= < <= ==

## logische Operatoren

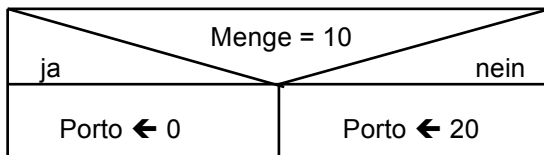
: && || !



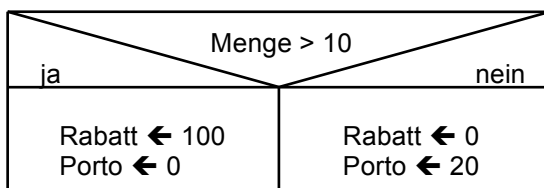
```
if (Menge > 10)
    Preis = Preis - 100;
```



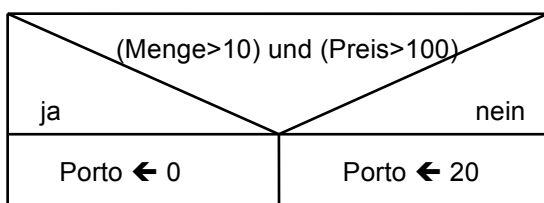
```
if (Menge >= 10) {
    Rabatt = 100;
    Preis = Preis - Rabatt;
}
```



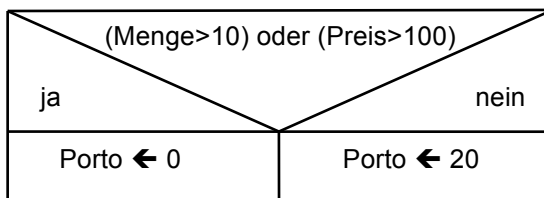
```
if (Menge == 10)
    Porto = 0;
else
    Porto = 20;
```



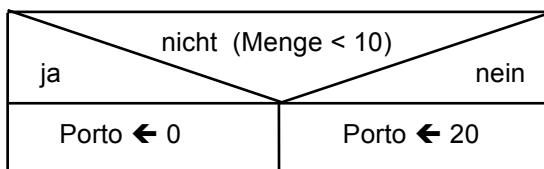
```
if (Menge > 10) {
    Rabatt = 100;
    Porto = 0;
}
else {
    Rabatt = 0;
    Porto = 20;
}
```



```
if ((Menge > 10) && (Preis > 100))
    Porto = 0;
else
    Porto = 20;
```



```
if ((Menge > 10) || (Preis > 100))
    Porto = 0;
else
    Porto = 20;
```



```
if (! (Menge < 10))
    Porto = 0;
else
    Porto = 20;
```