

5 Kontrollstrukturen

Allgemeine Strukturen (wie z.B. in Java auch):

```
if (Ausdruck) {} // Bedingter Block
```

```
if (Ausdruck) {}  
else {} // Verzweigung
```

```
if (Ausdruck1) {} // Mehrfache Verzweigung  
else if (Ausdruck2) {}  
else if (Ausdruck3) {}  
...  
else
```

```
while (Ausdruck) {} // Schleife
```

```
do {} while (Ausdruck); // Schleife, Ausführung mindestens einmal
```

```
for (init; Verweilbedingung; step) {} // Schleife
```

```
switch(x) { // Mehrfachverzweigung: x=='a', x=='b', sonst  
  case 'a' :  
    statement(s);  
    break; // optional // Verlassen des gesamten Switch-Konstrukts  
  case 'b' :  
    statement(s);  
    break; // optional  
  default : // optional // Wenn kein anderer Fall ...  
    statement(s);  
}
```

Mittels `break`-Anweisung können for-, while- und do-while-Schleifen sofort verlassen werden.

Mittels `continue`-Anweisung kann in Schleifen sofort zur Prüfung der Schleifenbedingung gesprungen werden, das gilt für sämtliche Schleifen: for, while und do-while.

Zusätzlich zu den bekannten Kontrollstrukturen gibt es in den bedingten Ausdruck:

```
ergebnis = (test) ? val1 : val2;
```

test ist ein boolscher Wert. Wenn **test** wahr ist ($\neq 0$), wird `val1`, ansonsten `val2` an `ergebnis` zugewiesen. `val1`, `val2` und `ergebnis` müssen entsprechend passende Datentypen haben. Es gibt ja keinen boolschen Datentyp. Für `test` wird geprüft ob dessen Wert ungleich 0 ist.