

# Struktogramm

Ein Struktogramm ist ein Diagrammtyp zur Darstellung von Programmentwürfen bzw. -abläufen.

## Anweisungen

Jede Anweisung wird in einen rechteckigen Strukturblock geschrieben. Anweisungen sind z.B. Variablendefinitionen, -zuweisungen, Funktionsaufrufe oder Ausgaben.

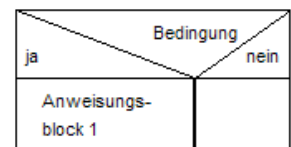
## Verzweigungen

Verzweigungen können Anweisungen Bedingungen auferlegen. Die Anweisungen werden dann nur ausgeführt, wenn die Bedingung ein bestimmtes Ergebnis (true, false) hat. Verzweigungen können beliebig verschachtelt werden. Sie werden dann wie Anweisungsblöcke einfach unter das gewünschte Ergebnis gesetzt.

Man unterscheidet folgende Verzweigungen:

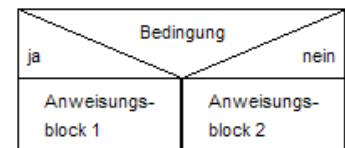
- **1 möglicher Block (If):**

Nur wenn die Bedingung zutrifft (wahr ist), wird der Anweisungsblock 1 durchlaufen. Ein Anweisungsblock kann aus einer oder mehreren Anweisungen bestehen. Trifft die Bedingung nicht zu (falsch), wird der Durchlauf ohne eine weitere Anweisung fortgeführt (Austritt unten).



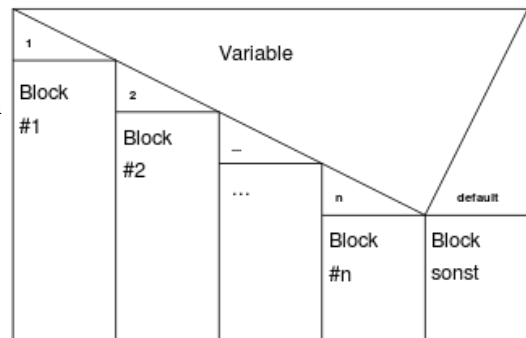
- **2 mögliche Blöcke (If-Else):**

Wenn die Bedingung zutrifft (wahr ist), wird der Anweisungsblock 1 durchlaufen; trifft die Bedingung nicht zu (falsch), wird der Anweisungsblock 2 durchlaufen (if-else). Ein Anweisungsblock kann aus einer oder mehreren Anweisungen bestehen.



- **Switch-Case-Verzweigung:**

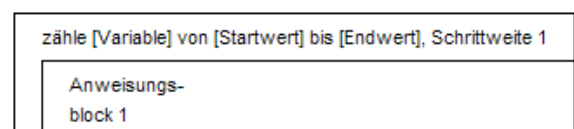
Besonders bei mehr als drei abzutestenden Bedingungen geeignet. Der Wert von „Variable“ kann bedingt auf Gleichheit wie auch auf Bereiche (größer/kleiner bei Zahlen) geprüft werden und der entsprechend zutreffende „Fall“ (case) mit dem zugehörigen Anweisungsblock wird durchlaufen. Unter default kann ein Standard-Fall angegeben werden.



## Schleifen

### Iteration Symbol (For-Schleife):

Wiederholungsstruktur, bei der die Anzahl der Durchläufe festgelegt ist (Endwert - Startwert)



Entspricht:

```
for (Variable := Startwert; Startwert < Endwert; Variable += Schrittweite) {  
    /* Anweisungsblock 1 */  
}
```

# Struktogramm

## Beispiel: BMI-Rechner

1. Körpergröße (in kg) und Gewicht (in cm) vom Nutzer einlesen
2. BMI-Wert berechnen:

$$BMI = \frac{\text{Gewicht (kg)}}{\text{Größe (m)}^2}$$

3. BMI mit Nachricht ausgeben:

| BMI        | Nachricht                  | Farbe |
|------------|----------------------------|-------|
| Bis 24,9   | „Sie haben Normalgewicht.“ | grün  |
| Ab 25,0    | „Sie sind zu dick.“        | rot   |
| Unter 18,5 | „Sie sind zu dünn.“        | rot   |

### Umsetzung im Struktogramm:

1. **Variablen definieren** (erster Anweisungsblock)
2. **Variablen zuweisen**  
(Anweisungsblöcke 2 bis 6),  
Standardfälle definieren  
→ **color** ← „green“  
→ **message** ← „Sie haben Normalgewicht.“
3. **Abfragen:**  
**bmi** >= 25 (erster Verzweigungs-Block):  
Ja?  
→ **color** ← „rot“  
→ **message** ← „Sie sind zu dick.“  
Nein?  
**bmi** <= 18.4:  
Ja?  
→ **color** ← „rot“  
→ **message** ← „Sie sind zu dünn.“

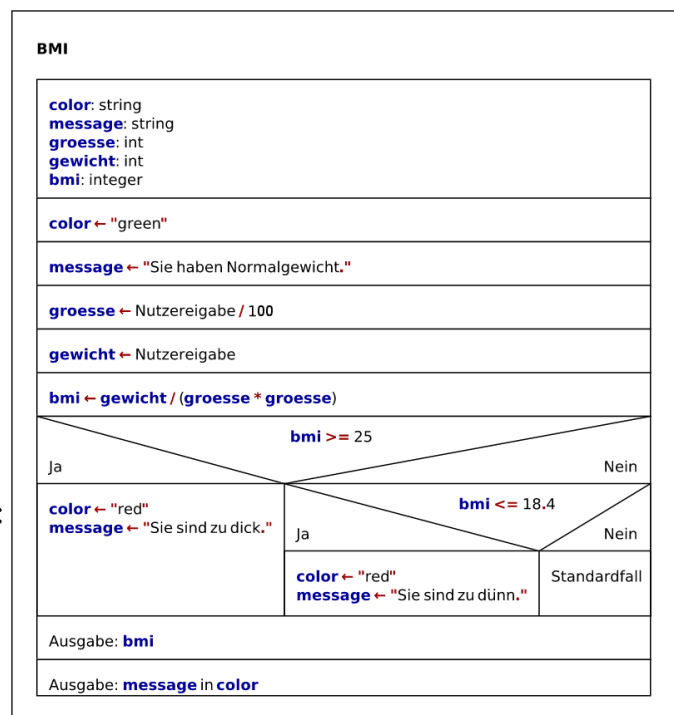


Abbildung 1: Struktogramm am Beispiel eines BMI-Rechners

4. **Ausgaben:**  
Ausgabe **bmi** (vorletzter Anweisungsblock)  
Ausgabe **message** in **color** (letzter Anweisungsblock)