

# Merkblatt - Grundbausteine eines Algorithmus

Zum Programmieren einer Software benötigst Du ein Programmier-Programm.

Jedes Programmier-Programm hat seine eigene Programmier-Sprache wie z. B. **Java**, **C++**, **Python**, **Visual Basic**, **SQL** usw.

Programmiersprachen sind z. B.:



Die **Schreibweise** einer Programmier-Sprache nennst Du **Syntax**.

Ist die Syntax fehlerhaft, dann sprichst Du von einem **Bug** (deut. Fehler). Die **Fehlerbehebung** nennst Du **Debuggen**.

Eine Programmier-Sprache besteht aus vielen Algorithmen. Algorithmen enthalten sogenannte **Grundbausteine** wie **Anweisung**, **Wiederholung (Schleife)**, **Variable (Parameter)** und **Bedingung**. Mehrere einzelne Anweisungen hintereinander nennst Du **Sequenz**.

## Abbildungen von Grundbausteinen im Programmier-Programm ROBOT KAROL.

### 1.) Anweisungen

PROGRAMM-EINGABE

```
1 Hinlegen
2 Schritt
3 Hinlegen
4 Schritt
5 Hinlegen
6 Schritt
```

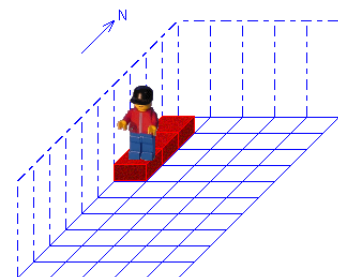
mehrere Anweisungen untereinander nennst Du **Sequenz**

STRUKTOGRAMM

Hauptprogramm

Hinlegen
Schritt
Hinlegen
Schritt
Hinlegen
Schritt

ROBOT KAROL



### 2.) Wiederholung/Schleife mit einer Variable/Parameter (z. B. 3)

PROGRAMM-EINGABE

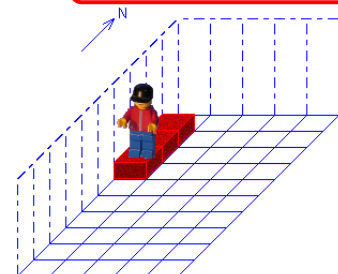
```
1 wiederhole 3 mal
2 Schritt
3 Hinlegen
4 *wiederhole
```

STRUKTOGRAMM

Hauptprogramm

wiederhole 3 mal
Schritt
Hinlegen

ROBOT KAROL



### 3.) Wiederholung/Schleife mit einer Bedingung z. B. solange

PROGRAMM-EINGABE

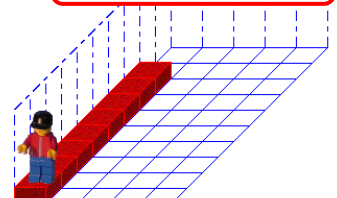
```
1 wiederhole
2 Schritt
3 Hinlegen
4 *wiederhole solange NichtIstWand
```

STRUKTOGRAMM

Hauptprogramm

Schritt
Hinlegen
wdh. solange NichtIstWand

ROBOT KAROL



#### 4.) Wiederholung/Schleife mit einer Bedingung z. B. bis

##### PROGRAMM-EINGABE

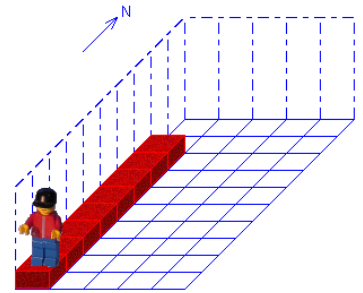
```
1 wiederhole
2 Schritt
3 Hinlegen
4 *wiederhole bis IstWand
```

##### STRUKTOGRAMM

###### Hauptprogramm

Schritt
Hinlegen
<b>wdh. bis</b> IstWand

##### ROBOT KAROL



#### Anweisungen und Bedingungen selbst schreiben!

##### PROGRAMM-EINGABE

```
{selbstdefinierte
Anweisung erstellen!}

Anweisung LegeZiegel
  wiederhole 3 mal
  Hinlegen
  *wiederhole
  *Anweisung

{Hauptprogramm!}

wiederhole
LegeZiegel
*wiederhole solange NichtIstWand
```

##### STRUKTOGRAMM

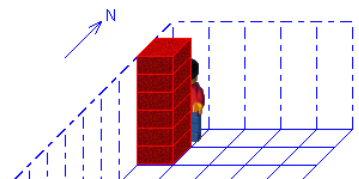
###### Hauptprogramm

LEGEZIEGEL
<b>wdh. solange</b> NichtIstWand

Anw.: LEGEZIEGEL

<b>wiederhole 3 mal</b>
Hinlegen

##### ROBOT KAROL



##### PROGRAMM-EINGABE

```
{selbstdefinierte
Bedingung erstellen!}

Bedingung IstZiegelLinks
  falsch
  RechtsDrehen
  wenn IstZiegel dann
  wahr
  *wenn
  LinksDrehen
  *Bedingung

{Hauptprogramm!}

Schritt
wiederhole solange IstZiegelLinks
Schritt
*wiederhole
```

##### STRUKTOGRAMM

###### Hauptprogramm

Schritt
<b>wdh. solange</b> IstZiegelLinks()
Schritt

###### Bedingung: IstZiegelLinks()

<b>falsch</b>
RechtsDrehen
<b>IstZiegel?</b>
<b>w</b>
<b>wahr</b>
LinksDrehen
<b>f</b>

##### ROBOT KAROL

