



Vorlesung „Grundlagen der Informatik und Programmierung 1“

SPRACHELEMENTE

Programmgerüst, Variablen und Operatoren

Prof. Dr. Tom Vierjahn

Visual Computing (<https://vc.w-hs.de>)

Fachbereich Wirtschaft und Informationstechnik – Campus Bocholt



Wintersemester 2020/21



Veröffentlicht unter der Creative-Commons-Lizenz

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

main.c

```
#include <stdio.h>

void main()
{
    printf("GIP 1");
}
```

Programmgerüst

etwas erweitert

main.c

```
#include <stdio.h>

void main()
{
    printf("Rechteckfläche: %d \n", 3 * 5);
}
```

Variablen

Definition, Wertzuweisung

main.c

```
#include <stdio.h>

void main()
{
    int side_a;
    int side_b;
    int area;

    side_a = 3;
    side_b = 5;
    area = side_a * side_b;

    printf("Rechteckfläche: %d \n", area);
}
```

Variablen

Definition mit Initialisierung – ganze Zahlen

main.c

```
#include <stdio.h>

void main()
{
    int side_a = 3;
    int side_b = 5;
    int area = side_a * side_b;

    printf("Rechteckfläche: %d \n", area);
}
```

Variablen

Definition mit Initialisierung – Gleitkomma-Zahlen

main.c

```
#include <stdio.h>

void main()
{
    float side_a = 3.1;
    float side_b = 5.2;
    float area = side_a * side_b;

    printf("Rechteckfläche: %f \n", area);
}
```

Definition: Variable

Eine **Variable** ist ein Speicherbereich, der einen **Namen** hat, und der Daten eines bestimmten **Typs** aufnehmen kann. Das hinterlegte Datum heißt **Wert** der Variablen.

```
int side_a = 3;
```

```
float side_b = 5.1;
```

Operator	Verwendung	Bedeutung
+	$x + y$	Addition $x + y$
-	$x - y$	Subtraktion $x - y$
*	$x * y$	Multiplikation $x \cdot y$
/	x / y	Division $\frac{x}{y}, y \neq 0$
%	$x \% y$	Rest aus ganzzahliger Division $\frac{x}{y}, y \neq 0$

Wichtig: *, / und % binden stärker als + und -.

Verwenden Sie im Zweifel Klammern.

- ▶ Ganze Zahlen und Gleitkommazahlen können gemischt werden.
- ▶ Es wird so lange wie möglich mit ganzen Zahlen gerechnet.
- ▶ Ab der ersten Gleitkommazahl, wird mit Gleitkommazahlen gerechnet.

Zuweisungen

Mit und ohne weitere(r) Operation.

- ▶ Zuweisung:
- ▶ ... mit weiterer Operation:
- ▶ ... oder kürzer:

Operator	Verwendung	Entsprechung
<code>+=</code>	<code>x += y</code>	<code>x = x + y</code>
<code>-=</code>	<code>x -= y</code>	<code>x = x - y</code>
<code>*=</code>	<code>x *= y</code>	<code>x = x * y</code>
<code>/=</code>	<code>x /= y</code>	<code>x = x / y</code>
<code>%=</code>	<code>x %= y</code>	<code>x = x % y</code>

Inkrement und Dekrement

Wir wollen zählen.

▶ Inkrement: `++` oder `++`

▶ ... oder kürzer: `++` oder `++`

▶ Dekrement: `--` oder `--`

▶ ... oder kürzer: `--` oder `--`

Inkrement und Dekrement

Wie unterscheiden sich die beiden Varianten?

► für alle:

```
int x = 1;
```

► Inkrement:

```
int c = x++;
```

$x =$, $c =$

```
int c = ++x;
```

$x =$, $c =$

► Dekrement:

```
int c = x--;
```

$x =$, $c =$

```
int c = --x;
```

$x =$, $c =$

- ▶ Es wird so lange wie möglich mit ganzen Zahlen gerechnet.

```
int a = 5;  
int b = 3;  
int c = a / b;
```

$c =$

- ▶ Nehmen wir float:

```
float c = a / b;
```

$c =$


- ▶ Ab der ersten Gleitkommazahl, wird mit Gleitkommazahlen gerechnet.

```
float c = ((float)a) / b;
```

$c \approx$

- ▶ Programmgerüst
- ▶ Variablen
- ▶ Arithmetische Operatoren
- ▶ Zuweisung
- ▶ Inkrement, Dekrement
- ▶ Typkonvertierung

Prof. Dr. Tom Vierjahn

▶  tom.vierjahn@w-hs.de

Visual Computing

▶  <https://vc.w-hs.de>

▶  VisualComputingWH

▶  Visual Computing WH

▶  @VisComputingWH

Westfälische Hochschule

Fachbereich Wirtschaft und Informationstechnik

Campus Bocholt



Veröffentlicht unter der Creative-Commons-Lizenz

Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)