

Berechnungen

Sinn und Zweck:

Berechnung sind das Herz jedes Programmes. Ein Programm das nichts berechnet, tut meist auch nichts sinnvolles. Berechnungen können unbemerkt ablaufen oder ziemlich offensichtlich, wenn man den z.B. den Taschenrechner in Windows benutzt.

Damit man etwas berechnen kann, muss man die Zahlen und die Ergebnisse irgendwo speichern. Dazu können wir unsere **Variablen** verwenden.

Die zum Rechnen notwendigen Rechenzeichen heißen in der Programmierung **Operatoren**.

Operatoren zum Rechnen in C:

*	Multiplikation
/	Division
%	Teilen mit Rest (Modulo)
+	Addition
-	Subtraktion

Diese Operatoren haben eine Rangfolge genauso wie in der Mathematik. Z.B. Punkt vor Strichrechnung beachtet der Computer ebenfalls.

Syntax: ErgebnisVariable = Ausdruck1 **Operator** Ausdruck2 ;

WICHTIG: dabei kann Ausdruck für eine **Variable** oder eine **Zahl** stehen

[illegible]

Hinweise zum Berechnen:

Zuerst wird immer die rechte Seite der **Zuweisung** (=) ausgeführt. Erst das komplette Ergebnis wird in der Variablen auf der linken Seite gespeichert.

Der Datentyp auf der linken Seite muss das Ergebnis auch speichern können!

```
Bsp:      iErgebnis = 3 / 4;           //FEHLER, 0,75 passt nicht in integer

          fErgebnis = 3.0 / 4.0:       //geht, weil float ja Kommazahlen speichert
```



Name:

AB No.:

Klasse:

Übungen zu Berechnungen

Übung 1:

Bei den folgenden Code-Ausschnitten, welches Ergebnis steht in fResult?

```
fResult = 0;
iVar = 18;
fVar = 0.5;
fHelp = 0.2;
iLoop = 1;
```

Code - Berechnung	fResult
fResult = iVar * 2;	
fResult = iVar / fVar - iLoop;	
fResult = fHelp * (fVar + 9.5);	
fResult = fHelp + 1 / fVar - iLoop;	
fResult = iLoop + 23*2 - 15/5 - 3*10;	
fResult = (iLoop + iVar) * ((fHelp+fHelp)/iVar)	

Übung 2:

Welchen Fehler begeht der Programmierer in diesem Programm? Begründe deine Antwort:

```
int iResult;
int iZahl1;
int iZahl2;
iZahl1 = 20;
iZahl2 = 2;
iResult = iZahl1 / iZahl2;
printf("Das Ergebnis der Division lautet %d", iResult);
iZahl2 = iZahl2 + 1;

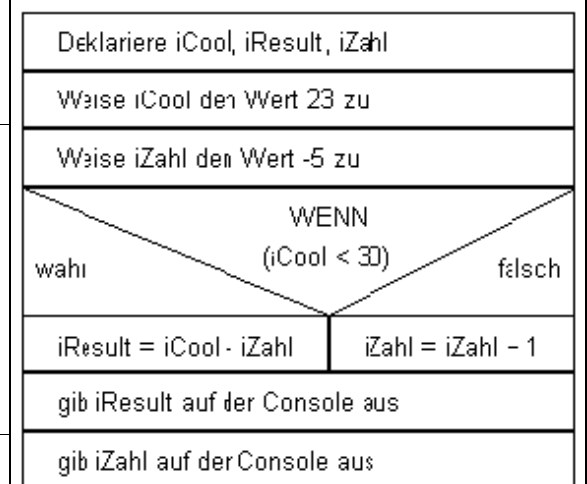
iResult = iResult - iZahl1 / iZahl2;

printf("Das Ergebnis lautet %d", iResult);
```

Übung 2:

Implementiere das folgende Struktogramm:

Übung 3



Übung 4:

Schreibe ein Programm, welches zwei Variablen fNumber1 und fNumber2 benutzt. Diese Variablen werden auf zwei beliebige Werte gesetzt. Anschließend gibt das Programm das Produkt, die Summe, die Differenz und den Quotienten der beiden Zahlen aus. Speichere dazu das Ergebnis in jeweils in einer Variablen fResult.

Entwerfe zunächst ein Struktogramm und lass es vom Lehrer kontrollieren!