



Name:

AB No.:

Klasse:

Wiederholungen von Befehlen (hier: For-Schleife)

Sinn und Zweck:

Wenn ein Programm ausgeführt wird, dann gibt es Befehle oder „Gruppen“ von Befehlen, die in derselben Reihenfolge mehrfach hintereinander ausgeführt werden müssen. Damit man nicht jeden Befehl immer wieder erneut schreiben muss, benutzt man in Struktogrammen sogenannte Wiederholungen bzw. sogenannte **Iterationen**. (siehe altes Arbeitsblatt!)

Wenn bereits VORHER bekannt ist, wie häufig die Schleife durchlaufen werden muss (z.B. wenn man von 1 bis 20 zählen soll), **dann** benötigt man eine **spezielle kopfgesteuerte Schleife mit der man VORHER die Anzahl Durchläufe angibt**.

Für solche kopfgesteuerten Schleifen mit gibt es einen Befehl, der beim Programmieren das das Wiederholen bestimmter Befehle veranlasst.

Syntax:

Befehl: **for**

Bei kopfgesteuerten **for-Schleife** wird am Anfang eine sogenannte **Schleifen-Variable** (LoopVar) benötigt. Ebenfalls muß angegeben werden, wann die Schleife beendet werden soll, die sogenannte **Abbruch-Bedingung**. Damit sich die Variable nach jedem Durchlauf ändert, muß als letztes noch angegeben werden, um **wieviele** sich die Schleifen-Variable nach jedem Durchlauf ändern soll, der sogenannte **Inkrement**.

ACHTUNG: Die Schleifen-variable muss deklariert sein!

Wenn die Bedingung dann irgendwann nicht mehr erfüllt ist, dann wird der Inhalt der Schleife nicht erneut ausgeführt.

<i>Struktogramm FOR-Schleife</i>	C
<div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> LoopVar=SW ; Bedingung ; Inkrement <div style="border: 1px solid black; padding: 5px; margin: 5px auto; width: fit-content;"> Anweisungen die wiederholt werden sollen </div> </div>	<pre>for (LoopVar=StartWert ; Bedingung ; Inkrement) { Anweisungen die wiederholt werden } </pre>

Beispiele:

Beispiel 1	Beispiel 2
<pre>int iLoop; for (iLoop=0 ; iLoop<=10 ; iLoop=iLoop+1) { printf("Jetzt ist i so gross %d",iLoop); newline(); printf("yep"); newline(); } </pre>	<pre>int iLoop; for (iLoop=5 ; iLoop<17 ; iLoop=iLoop+2) { printf("Hier ist i so gross %d",iLoop); newline(); } </pre>
<p>Diese Schleife zählt von 0 bis 10 in 1er Schritten</p> <p>iLoop hat nach Durchlauf den Wert 11</p>	<p>Diese Schleife zählt von 5 bis 17 in 2er Schritten.</p> <p>iLoop hat nach Durchlauf den Wert 17</p>



Name:

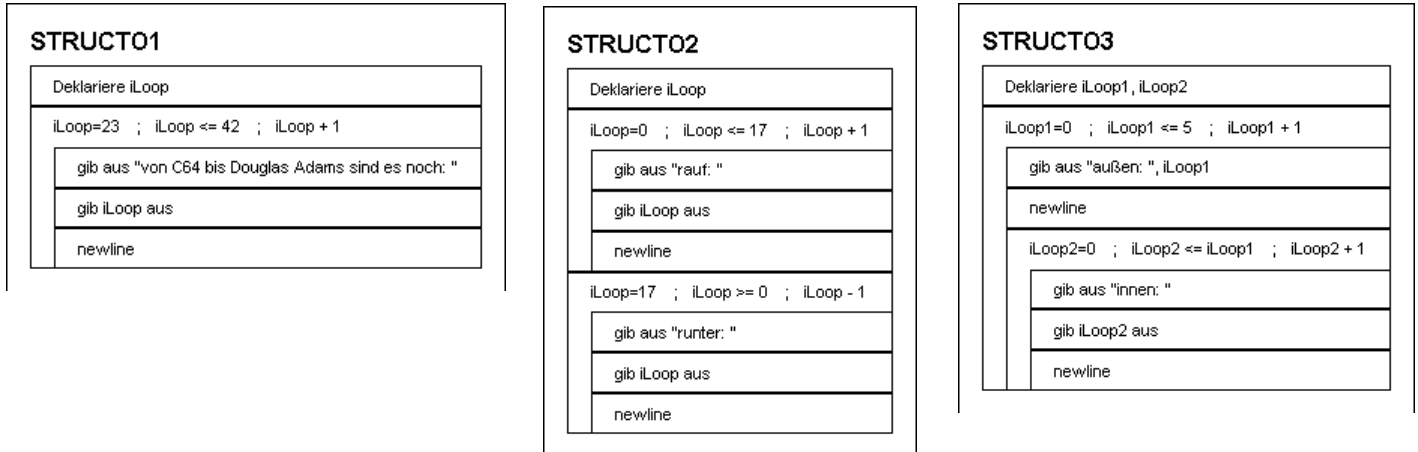
AB No.:

Klasse:

Übungen zu Wiederholungen

Übung 1 bzw Übung 2 bzw. 3:

Erstelle ein jeweils ein **eigenes Programm** gemäß den folgenden Struktogrammen:



Übung 4:

Schreibe ein Programm, welches den Countdown des Spaceshuttles nachzählt. Der Countdown beginnt bei 10 und endet bei 1. Danach kommt der übliche Spruch „Ignition and LIFT OFF“.

Mit dem Befehl `sleep(1)` kann man das Programm für 1 Sek. anhalten, 2 = 2 Sekunden, etc.

Übung 5:

Schreibe jeweils ein Programm, welches auf der Konsole die folgenden Ausgaben erzeugt. Benutze dazu `for`-Schleifen:

a)	b)	c)	d)
Hallo 23	13	1	012345
Hallo 24	11	12	01234
Hallo 25	9	123	0123
Hallo 26	7	1234	012
Hallo 27	5	12345	01
Hallo 28	3	123456	0

Übung 6:

a) Schreibe ein Programm, bei dem der Benutzer die Zahlen `iYAbschnitt` und `iSteigung` eingibt. Anschließend erstellt das Programm eine Wertetabelle für die lineare Funktion:

$$f(iX) = iSteigung * iX + iYAbschnitt$$

Hinweis: `iX` durchläuft mit Hilfe einer `for`-Schleife die Werte von 0 bis 10. Das Ergebnis der Rechnung `iSteigung * iX + iYAbschnitt` muss zwischengespeichert werden, damit es in der Schleife ausgegeben werden kann.

b) Ändere das Programm aus a) so ab, dass die Wertetabelle von `iX=-5` bis `iX=5` berechnet wird. Formatiere die Ausgabe möglichst ordentlich.