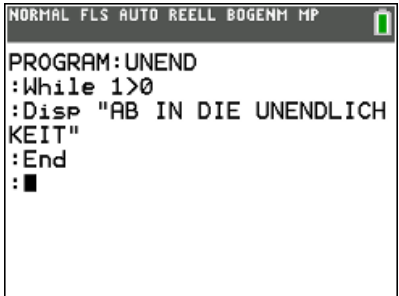
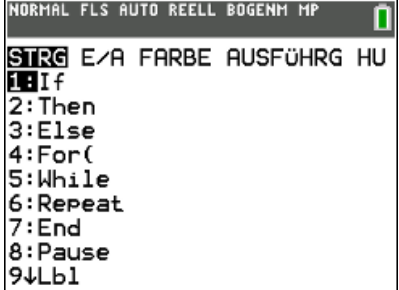




Lektion 4: Schleifen	Übung 1: Die For(...) - Schleife
<p>In der ersten Übung der Lektion 4 geht es um das Schleifenkonzept und speziell um die <b>For(...)</b> – Schleife.</p>	<p><b>Lernziele:</b></p> <ul style="list-style-type: none"> <li>• Das Schleifenkonzept verstehen.</li> <li>• Verwendung der <b>For(...)</b> – Schleife, um eine Wertetabelle zu erzeugen.</li> </ul>
<p><b>Hinweis:</b> Es gibt drei grundlegende Schleifen im TI-Basic: <b>For</b>, <b>While</b> und <b>Repeat</b>. In einer Schleife wird eine Gruppe von Anweisungen immer wieder ausgeführt, solange bis ein Zähler einen bestimmten Grenzwert erreicht hat (<b>For(...)</b> – Schleife) oder solange bis eine bestimmte Bedingung erfüllt oder nicht erfüllt ist (<b>While</b> und <b>Repeat</b>). In dieser Lektion wird in jeder Übung eine dieser Schleifen vorgestellt. Natürlich kann man auch mit den einfachen <b>Lbl</b> - und <b>Goto</b> – Befehlen eine Schleife erzeugen, aber das Programm sieht dann nicht gut aus und es können sich leicht Programmierfehler einschleichen. Deshalb vermeiden wir diese Befehle wo immer es geht. Allerdings wird der Befehl <b>Lbl</b> durchaus noch verwendet und muss sogar zwingend in Zusammenhang mit dem Befehl <b>Menu</b> verwendet werden. Aber der Befehl <b>Goto</b> ist für Bedingungen und Schleifen überflüssig.</p> <p>Programme werden dadurch kompliziert, dass man meist Kontrollstrukturen und Schleifen in einem Programm mischen muss, um den Algorithmus durchführen zu können. Allerdings macht das erst den SPASS am Programmieren aus!</p>	
<p><b>Schleifen</b></p> <p>Eine Schleife ist eine Methode, um eine Gruppe von Befehlen zu wiederholen. Alle Programmiersprachen haben mindestens eine Schleifenstruktur. Mit einer Schleife geht man im Programm ein paar Schritte zurück zu einer vorhergehenden Stelle und führt das Programm von da erneut aus. TI-Basic hat drei verschiedene Typen von Schleifen.</p> <p>Es kann passieren, dass eine Schleife nicht endet (unendliche Schleife). Um ein laufendes Programm zu unterbrechen, drückt man <b>ON</b>. Dann hat man die Wahlmöglichkeiten „Verlassen“ und „Gehezu“. „Verlassen“ führt zurück auf den HOME-Bildschirm, „Gehezu“ öffnet den Programmeditor an der Stelle, wo das Programm angehalten wurde.</p>	 <p>An infinite loop. Why?</p>
<p>Die drei Schleifen in TI-Basic sind:</p> <p><b>For( ) ... End</b></p> <p><b>While</b> &lt;Bedingung ist wahr&gt; ... <b>End</b></p> <p><b>Repeat</b> &lt;bis Bedingung ist wahr&gt; ... <b>End</b></p> <p>Diese erste Übung behandelt die <b>For(...)</b> – Schleife.</p>	



#### Die For( ... ) - Schleife

Aufbau: **For**(Variable, Startwert, Endwert [, Schrittweite])  
 <Gruppe von Befehlen>  
**End**

Beispiel: **For**(A,1,10) Schrittweite kann entfallen  
**Disp** A  
**End**

```
NORMAL FLS AUTO REELL BOGENM MP
PROGRAM:ZAEHLER
:For(A,1,10)
:Disp A
:End
:█
```

#### Hinweis:

Der Befehl **For**( ) benötigt eine Variable, die Schleifenvariable, einen Startwert und einen Endwert, die durch Komma getrennt sind. Start- und Endwert können Variablen sein und die Befehlsgruppe kann beliebig viele Befehle umfassen, sollte aber keinen darunter haben, der die Schleifenvariable verändert. Die Schleife läuft vom Start- bis zum Endwert mit der Standard-Schrittweite 1.

```
NORMAL FLS AUTO REELL BOGENM MP
3
4
5
6
7
8
9
10
.....Fertig
```

**Hinweis:** Die Schleife endet, wenn der Endwert überschritten wird. Lässt man sich mit **Disp A** den Wert der Schleifenvariablen nach dem Durchlaufen der Schleife ausgeben, so ist ihr Wert im obigen Beispiel 11, nicht etwa 10!

#### For – Schleifen mit anderen Schrittweiten

Das optionale vierte Argument im Befehl **For**(... ) ist die Schrittweite. Sie bestimmt, um welchen Wert die Schleifenvariable bei jedem Durchgang wächst. Der voreingestellte Wert ist 1.

```
NORMAL FLS AUTO REELL BOGENM MP
PROGRAM:ZAEHLER
:For(A,1,10,3)
:Disp A
:End
:
```

**Hinweis:** Ist die Schrittweite negativ, so muss der Startwert größer als der Endwert sein. Die Schleife endet, wenn die Schleifenvariable kleiner als der Endwert ist.

#### Beispiele:

**For(A,1,10,3)** beginnt mit **A=1**, dann wird bei jedem Schleifendurchlauf 3 zu **A** addiert. Die Schleife endet, wenn **A** größer ist als 10.

**For(B,10,0,-1)** zählt von 10 bis 0 herunter (negative Schrittweite).

```
NORMAL FLS AUTO REELL BOGENM MP
PRgmZAEHLER
1
4
7
10
.....Fertig
█
```



#### Programmieren mit For(...)

Es soll ein Programm geschrieben werden, dass die natürlichen Zahlen eines bestimmten Zahlbereiches und ihre Quadratzahlen anzeigt.

Der Benutzer gibt den Anfangs- und den Endwert des Bereiches ein. Bei der Darstellung wird ein kleiner Trick angewendet, denn um mit dem Befehl **Disp** zwei Zahlen in einer Zeile darzustellen wird eine **Liste** verwendet. Die Elemente der Liste stehen hinter dem Befehl **Disp** in geschweiften Klammern, getrennt durch ein Komma. Man erhält sie über  $\boxed{2nd} \boxed{[ ]}$  und  $\boxed{2nd} \boxed{[ ]}$

```

NORMAL FLS AUTO REELL BOGENM HP
PROGRAM: PARABEL
:ClrHome
:Input "UNTERGRENZE?",U
:Input "OBERGRENZE?",O
:For(A,U,O)
:Disp {A,A^2}
:End
:

```

**Hinweis:** Die Darstellung als Liste ist eine übliche Methode, um mehr als eine Zahl in einer Zeile auf dem HOME-Bildschirm darzustellen.

Nun kann das Programm gestartet werden.

Sollte das Programm zu schnell ablaufen, kann man an einer Stelle der Schleife den Befehl **Pause** einfügen.

#### Aufgabe:

Mit **If ... Then... End** soll erreicht werden, dass alle 5 Paare von Antworten eine Pause eingelegt wird. Dazu wird die Teilbarkeitsanweisung aus der letzten Einheit verwendet.

#### Beispielantwort:

```

:If (A-U+1)/5=int((A-U+1)/5)
:Pause "DRUECKE ENTER"

```