



Lektion 3: Programmieren mit Bedingungen	Übung 1: Bedingungen und der Befehl If ...
--	--

In der ersten Übung der Lektion 3 geht es um Bedingungen und eine Einführung in den If – Befehl des TI Basic.	Lernziele: <ul style="list-style-type: none"> Verwendung von Bedingungen. Verwendung der einfachsten Form des If ... - Befehls, um abhängig von einer Bedingung einen weiteren Befehl auszuführen.
--	--

If...Then – Befehle werden verwendet, um eine Gruppe von Befehlen nur dann auszuführen, wenn eine Bedingung wahr ist. Aber was ist eine *Bedingung*?

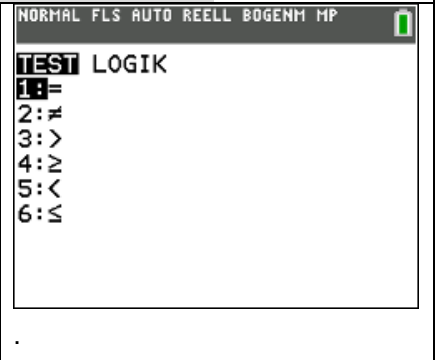
Hinweis: Relationale und logische Operatoren haben untereinander eine *Rangfolge*. Sie sind auch in die Rangfolge der arithmetischen Operationen eingeordnet und können mit ihnen zusammen verwendet werden.

Die vollständige Liste der TI-Basic - Rangfolge:

Rang	Funktionen
1	Funktionen, die vor ihrem Argument stehen wie z.B. $\sqrt{\quad}$ oder $\sin(\quad)$ mit Ausnahme der Negation
2	Funktionen, die ihrem Argument folgen wie z.B. \square^2 oder $!$.
3	\wedge und $\times\sqrt{\quad}$
3.5	Die Negation nicht(
4	nPr und nCr
5	Multiplikation, Division und implizite Multiplikation
6	Addition und Subtraktion
7	Die relationalen Operatoren =, \neq , <, >, \leq , \geq
8	Der logische Operator und
9	Die logischen Operatoren oder und xoder
10	Umwandlungen wie z.B. $\blacktriangleright F \blacktriangleleft D$

Bedingungen und das Menü [TEST]

Bedingungen sind Ausdrücke, die **entweder wahr oder falsch** sind; andere Möglichkeiten gibt es nicht. Werden sie ausgewertet, so ergibt **wahr** den Wert **1**, **falsch** den Wert **0**. Die relationalen und die logischen Operatoren findet man im Menü [TEST] (2nd [MATH]). Das Menü **TEST** enthält die **relationalen Operatoren**, das Menü **LOGIK** die **logischen**. Das **Gleichheitszeichen =** wird hier im Sinne einer Bedingung eingesetzt und nicht als Zuweisung wie sonst üblich.



Beispiele für Bedingungen

$3 > 5$ $XY > 0$ $X = 5$ oder $Y = 5$
 $X + 4 > X$ $B^2 - 4AC = 0$ $X/2 = \text{int}(X/2)$
 X oder Y $X > 0$ und $Y > 0$ nicht($X > 0$)





Bedingungen auf dem HOME-Bildschirm

Man kann Bedingungen auch auf dem HOME-Bildschirm verwenden, um ihre Wirkungsweise zu untersuchen.

Dabei steht die **1** für *wahr* und die **0** für *falsch*.

Hinweis: Verwendet man eine Variable in einer Bedingung, so rechnet der Taschenrechner mit dem aktuellen Wert der Variablen.

```

NORMAL FLS AUTO REELL BOGENM MP
3>5
----- 0
3≠5
----- 1
X+1=X
----- 0
X+1>X
----- 1

```

Hinweis: In dieser Anleitung wird zunächst der *einfache If* – Befehl eingeführt, da er eine besonders einfache Möglichkeit darstellt, wenn nur ein Befehl in Abhängigkeit von einer Bedingung ausgeführt werden soll. Allerdings sollte der später behandelte vielseitigere Befehl **If...Then** bevorzugt werden, da dadurch das Programm leichter lesbar wird. Beim *einfachen If* – Befehl wird nur die unmittelbar folgende Anweisung durchgeführt, wenn die Bedingung wahr ist; andernfalls wird sie übersprungen.

Ein erstes Programm mit dem einfachen If... - Befehl

If findet man im Menü **PRGM** STRG, '>' wie oben beschrieben.

Das Programm lautet

:Prompt A

:If A>0

:Disp "A IST POSITIV"

:Disp "A IST NICHT POSITIV"

```

NORMAL FLS AUTO REELL BOGENM MP
PrgmBED1
A=?5
A IST POSITIV
A IST NICHT POSITIV
----- Fertig

```

Lässt man das Programm mehrere Male laufen, so stellt man folgendes fest:

Stimmt die Bedingung **A>0**, so wird der auf **If** folgende Befehl ausgeführt, andernfalls einfach übersprungen. Aber der Befehl, der die Anzeige „A IST NICHT POSITIV“ erzeugt, wird stets ausgeführt, was nicht richtig ist. Der Fehler wird in einem weiteren Schritt korrigiert.

Der einfache **If** – Befehl ist eine sehr kurze Möglichkeit, um einen Befehl zu überspringen, wenn die dazugehörige Bedingung falsch ist.

Verbesserung des Programmes

Der Fehler kann korrigiert werden, indem man ein weiteres **If...** einfügt:

1. Der Cursor wird auf das zweite **Disp** bewegt.
2. Mit **[INS]** und **[ENTER]** wird eine Leerzeile eingefügt.
3. In diese Zeile kommt der Befehl **If A<0**.

```

NORMAL FLS AUTO REELL BOGENM MP
PROGRAM:BED1
:Prompt A
:If A>0
:Disp "A IST POSITIV"
:If A<0
:Disp "A IST NICHT POSITIV"
"
:

```

Nun kann man das Programm mit verschiedenen Werten für **A** ausprobieren, auch mit dem Wert **0**. Wenn sich herausstellt, dass das Programm noch nicht richtig funktioniert, so sollte man sich Gedanken über eine weitere Fehlerbehebung machen.