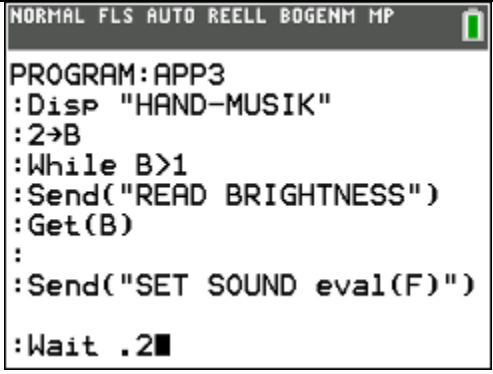
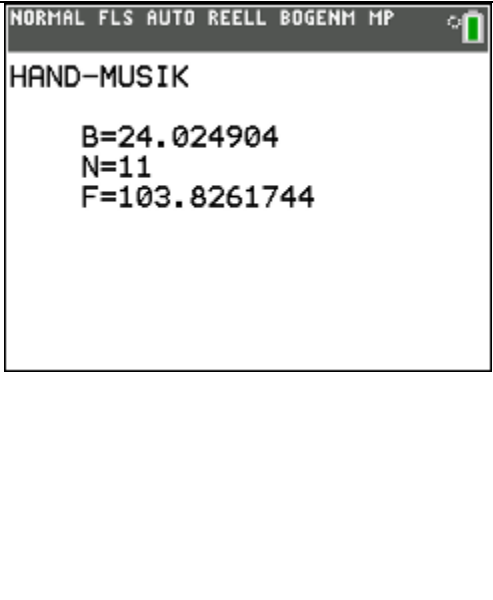




Lektion 3: Helligkeit, If und While	Anwendung: Musik mit der Hand
<p>In dieser Anwendung soll ein Programm entstehen, mit dem Tonfrequenzen durch die Änderungen der Helligkeit gesteuert werden, die ein Lichtsensor misst. Das kann man erreichen, indem man ihn mit der Hand abschattet – also Musik mit der Hand!</p>	<p>Lernziele:</p> <ul style="list-style-type: none"> • Schreiben eines Programms, das Helligkeitswerte in Tonfrequenzen umsetzt. • Wiederholung der Frequenzen der Musiknoten und des Prinzips der zwölften Wurzel aus 2.
<p>Es soll ein Programm geschrieben werden, dass die Helligkeit des Lichtsensors (BRIGHTNESS) ausliest und einen von dieser Helligkeit abhängigen Ton spielt. Dabei gibt es für den Ton zwei Möglichkeiten:</p> <ul style="list-style-type: none"> • Es wird einfach eine Frequenz des Hörbereiches gespielt (etwa 100Hz – 1000Hz). • Es wird eine Musiknote gespielt, so wie man sie z.B. auf einem Klavier oder einem anderen Instrument findet. <p>Die erste Möglichkeit erzeugt Töne, die man am ehesten mit "Geräusch" umschreiben kann. Mit der zweiten Möglichkeit erhält man Töne, die wie Musik klingen, aber die Mathematik ist etwas komplexer.</p> <p>Mit diesem Programm verhält sich der TI-Innovator™ Hub wie ein <i>Theremin</i> (1920 erfundenes elektronisches Instrument, das berührungslos gespielt wird).</p>	
<p>Anmerkungen zum Programm:</p> <ol style="list-style-type: none"> 1. Ein neues Programm mit dem Namen APP3 wird angelegt. 2. Die Befehle ClrHome and Disp werden hinzugefügt, damit die Überschrift wie abgebildet angezeigt werden kann. 3. Eine While – Schleife wird hinzugefügt, in der die Helligkeit ausgelesen und in einer Variablen B gespeichert wird. 4. Hinzugefügt werden muss auch noch der Befehl, mit dem der Ton auf den eingebauten Lautsprecher ausgegeben wird. Achtung! Für den Ton wird die Variable F verwendet; es fehlt noch die Umrechnung von B in F! 	 <pre> NORMAL FLS AUTO REELL BOGENM MP PROGRAM:APP3 :Disp "HAND-MUSIK" :2→B :While B>1 :Send("READ BRIGHTNESS") :Get(B) : :Send("SET SOUND eval(F)") :Wait .2 </pre>
<p>Das Programm muss noch um die fehlenden Programmzeilen ergänzt werden, mit denen die Helligkeit in einen Ton oder einen musikalischen Ton umgewandelt wird.</p> <p>Für einfache Töne hat sich der Frequenzbereich von 100 Hz – 1000 Hz als sinnvoll erwiesen (man sollte aber auch einmal einen anderen Bereich ausprobieren).</p> <p>Bei Musiknoten sollte man den Bereich bei a (55 Hz) beginnen lassen und z.B. die nachfolgenden 50 Töne verwenden (s. Lektion 2, Übung 3, Programm TON2 – hier werden die 12 Töne einer Oktave gespielt).</p> <p>Bei Musiknoten muss man die Helligkeit in ganze Zahlen umwandeln. Man kann entweder die Funktion int() oder die Funktion runde(,0) verwenden:</p> <p>int(X)→X erzeugt die größte ganze Zahl, die kleiner ist als X.</p> <p>runde(X,0)→X rundet X auf den nächsten ganzzahligen Wert.</p>	 <pre> NORMAL FLS AUTO REELL BOGENM MP HAND-MUSIK B=24.024904 N=11 F=103.8261744 </pre>

**Hinweis:**

Sollen nur Töne gespielt werden, muss der Bereich 0 bis 100 für **B** in den Bereich 100 bis 1000 für **F** konvertiert werden:

B	F
0	100
100	1000

Die Steigung der Geraden durch diese zwei Punkte ist

$$M = (1000-100)/(100-0) = 9$$

Damit erhält man als Geradengleichung

$$F = 9 \cdot B + 100$$

Um Musiknoten zu spielen, muss man die $F \cdot 2^{(1/12)}$ Eigenschaft der Notenintervalle nutzen. Startet man bei $a = 55$ Hz und will die nächsten 50 Noten verwenden, so bekommt die erste Note die Nummer 0 und die letzte die Nummer 49 (Programmierer beginnen stets mit der Null). Außerdem müssen die Notennummern ganze Zahlen sein, so dass man wieder `int()` oder `runde()` benutzen muss. Mit 2 Rechenschritten lässt sich die Umrechnung veranschaulichen:

$N = \text{int}(49B/100)$	-- Nummer der Note
$F = 55 \cdot 2^{(N/12)}$	-- Tonfrequenz der Note