



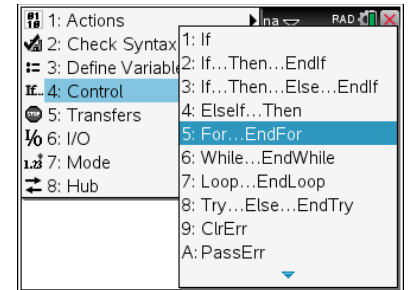
In de eerste les van unit 2 leer je de **For** loop (lus) kennen via een knipperend lampje en informatie die op het rekenmachinescherm wordt getoond.

#### Doelen:

- De **For** loop leren kennen
- Een lichtje laten knipperen.
- Gebruik van het **Disp** commando voor tekst en variabelen.

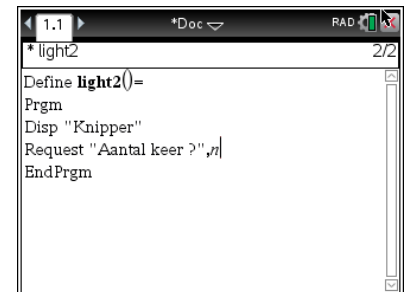
**Tip:** Er zijn 3 soorten 'loops' (lussen) in de TI-Nspire™ CX TI-Basic taal: **For**, **While**, en **Loop**. Iedere loop heeft zijn eigen **End...** aan het einde van de lus (loop). **EndFor** staat aan het einde van het **For** loop, etc. Het keyword **Stop** beëindigd het programma. Het keyword **Exit** springt naar het eerste commando na het einde van de loop en het **Cycle** springt naar het begin van een loop. We proberen deze commando's te vermijden maar **Exit** is noodzakelijk om uit een **Loop...EndLoop** lus te komen.

Laten we een programma gaan schrijven dat een LED een aantal keer laat knipperen. Met argumenten kun je bepalen hoe lang het LED-je aan en uit is. Dit programma introduceert de **For...EndFor** loop (lus).



#### Het programma:

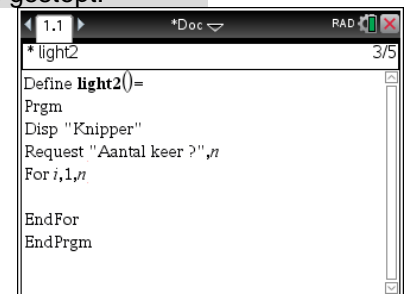
1. Begin een nieuw programma en noem het LIGHT2.
2. Voeg het commando **Disp** toe via **menu > I/O > 1:Disp**.
3. Schrijf tussen aanhalingstekens de tekst "Knipper" zoals hiernaast.
4. Voeg het commando **Request** toe via **menu > I/O > 1:Request**.
5. Schrijf tussen aanhalingstekens "Aantal keer ?"
6. Plaats daarna een komma en dan de variabele *n*



**Tip:** De **For** loop bestaat uit **For var, start, finish, <step>** en **EndFor** aan het einde. De loop body herhaalt automatisch als de variabele varieert van de beginwaarde (*start*) tot de eindwaarde (*finish*). De waarde *step* is altijd 1 als de waarde *step* niet is gespecificeerd. Aan het einde van de loop wordt de variabele *var* één *step* verhoogd. Dit betekent als de loop gaat van 1 tot 10 dat de variabele *var* gelijk is aan 11 na dat de loop is gestopt.

#### De For Loop toevoegen:

7. Kies de **For** loop via **menu > Besturing > 5:For...EndFor**.
  - Het begin van de For loop **For** , , en het einde **EndFor** worden in het programma geplakt met daar tussen in een lege regel voor de body van de loop.
8. Completeer de rest van commando **For** met **i,1,n**, tussen komma's
  - Dit commando betekent nu: "Voor i van 1 tot n in stappen van 1"



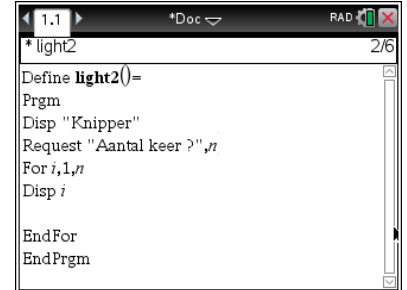
## 10 Minutes of Code

### TI-NSPIRE CX MET DE TI-INNOVATOR™ HUB

### UNIT 2: OEFENBLAD 1 DOCENTENHANDLEIDING

- Druk een aantal keer op enter in de loop body om lege regels in te voegen die we later invullen.
  - Het aantal lege regels is niet van belang. Je kunt er altijd meer invoegen. Lege regels beïnvloeden het programma niet.
  - De regels tussen For en EndFor wordt de 'loop body' genoemd. Dit is het deel van de code dat n keer wordt uitgevoerd door de **For** loop.


We willen dat het LED-je **n** keer aan en uit gaat. We willen dat de rekenmachine het knippernummer weergeeft. We starten de loop body met **Disp i**, de "loop"-variabele.



```
1.1 *Doc RAD 2/6
light2
Define light2()=
Prgm
Disp "Knipper"
Request "Aantal keer >",n
For i,1,n
Disp i
EndFor
EndPrgm
```

We gaan nu commando's toevoegen die er voor zorgen dat de LED aan en uit gaat, zoals hiernaast

- Voeg het commando **Send "SET LIGHT** toe vanuit het **HUB** menu.
- Voeg het woord **ON** toe vanuit het menu **HUB > SETTINGS** (je kunt het ook intikken).
- Voeg het commando **Wait** (in seconde) toe vanuit het **HUB** menu zodat het programma wacht voordat het het volgende commando stuurt.
  - In ons programma hebben we 1 seconde wachten gebruikt, maar je kunt elke waarde kiezen die je wilt (ook decimale waarden als 1.5).
- Voeg het volgende **Send** commando toe om de LED uit te doen en daarna nog een **Wait** commando.
- Voer het programma uit met **ctrl-R**



```
1.1 *Doc RAD 8/9
light2
Disp "Knipper"
Request "Aantal keer >",n
For i,1,n
Disp i
Send "SET LIGHT ON"
Wait 1
Send "SET LIGHTOFF"
Wait 1
EndFor
EndPrgm
```

*Zie het inspringen van de code in het voorbeeld. Dit helpt om programma's beter leesbaar te maken maar verandert niets aan de uitvoering van het programma.*

Een uitdaging: Voeg meer input commando's toe (voor de **For** loop) om de timing van de twee **Wait** commando's te variëren.

Laat het programma weer uitvoeren. Bekijk het knipperen en bekijk de waarden die worden getoond op het scherm van de rekenmachine.