



In deze derde les van module 3 leer je over een andere vorm van de **If**-opdracht in TI Basic en ontdek je het belang van het goed begrijpen van rekenprocedures en algoritmes.

#### Doelen:

- De structuur van de opdracht **If...Then...Else...End** begrijpen.
- De test voor 'geheel-zijn' (Is een waarde een geheel getal) leren
- Begrijpen wat een **Algoritme** is?

**Docenten Tip:** In deze les maken leerlingen kennis met het concept **algoritme**. Het begrijpen van het principe van een algoritme is van wezenlijk belang voor succesvol programmeren. Hoewel we niet te diep ingaan op een algoritmische benadering, is het idee van het hebben van een plan voordat het programma wordt geschreven cruciaal, omdat het schrijven van een programma allereerst vraagt om begrip van de taak en kennis van programmeeropdrachten en –commando's en hun doel. Dan pas kunnen we een algoritme ontwerpen om het probleem op te lossen. De drie vormen van de **If**-opdracht worden in deze les afgesloten met een bespreking van **If...Then...Else...End**. De syntax is belangrijk: elk van de vier sleutelwoorden verschijnt als een afzonderlijke opdracht in het programma en alleen de **If** kan door iets gevolgd worden (de *voorwaarde*).

#### Over If...Then...Else...End

In de vorige activiteit leerde je over de opdracht **If...Then...End**. Er zijn situaties waarin het nodig is om twee verschillende 'routes' te kunnen nemen afhankelijk van een voorwaarde. De structuur van deze nieuwe opdracht **If...Then...Else...End** is hetzelfde:

```
If <voorwaarde>  
Then  
  <'waar' blok>  
Else  
  <'onwaar' blok>  
End
```

*Opmerking:*

**Then, Else, en End** staan apart op een eigen regel.

Het <'waar' blok> bestaat uit de reeks opdrachten die uitgevoerd zullen worden als de <voorwaarde> waar is.

Het <Onwaar blok> bestaat uit de reeks opdrachten die uitgevoerd zullen worden als de <voorwaarde> onwaar is.

Daarom zal altijd een van deze twee blokken uitgevoerd worden.

### Programmeren met If...Then...Else...End

We gaan een programma schrijven dat vertelt of een ingevoerd getal een 'perfect' kwadraat is. Een 'perfect kwadraat' is het kwadraat van een geheel getal, zoals  $25 (5^2)$ . De methode die we hier gebruiken is het trekken van de wortel uit het getal en testen of dit een geheel getal is. Het volledige programma staat rechts.

*Opmerking:*

Gebruik de toets  $\sqrt{\quad}$  op het toetsenbord voor het wortel-symbool.

**If, Then, Else, en End** zijn allemaal te vinden in het CTL menu onder **p**.

De **int( )** -functie staat in het menu NUM onder **m**. Deze functie geeft als resultaat een geheel getal. Dus bijvoorbeeld:

$\text{int}(6.56) \rightarrow 6$     $\text{int}(9.999) \rightarrow 9$     $\text{int}(-2.01) \rightarrow -3$

Probeer je eigen **int( )**- voorbeelden uit op het beginscherm. Vergeet niet om de haakjes van de functie **int( )** en van de wortelfunctie te sluiten. Op de rekenmachine gebruik je alleen HOOFDLETTERS in je **Disp** -opdrachten. Je kunt kleine letters invoeren met behulp van **TI-Connect CE**.

```
NORM DRIJF AUTO REËEL RAD MN
PROGRAM: KWADRAAT
: WisHome
: Input "VOER EEN GETAL IN:
: .N
: If  $\sqrt{N}$ =int( $\sqrt{N}$ )
: Then
: Disp "DIT IS EEN PERFECT
KWADRAAT."
: Else
: Disp "DIT IS GEEN PERFECT
```

```
NORM DRIJF AUTO REËEL RAD MN
PROGRAM: KWADRAAT
".N
: If  $\sqrt{N}$ =int( $\sqrt{N}$ )
: Then
: Disp "DIT IS EEN PERFECT
KWADRAAT."
: Else
: Disp "DIT IS GEEN PERFECT
KWADRAAT"
: End
```

```
NORM DRIJF AUTO REËEL RAD MN
VOER EEN GETAL IN: 81
DIT IS EEN PERFECT KWADRA...
.....Klaar.....
```

**Docenten Tip:** Adviseer de leerlingen om de uitvoer van hun programma mooier te maken door het getal dat is ingevoerd in een zin af te beelden in plaats van alleen als het woord "DIT". Hiervoor is het nodig om **Output( )** te gebruiken in plaats van **Disp**.

### Algoritmes

Technieken zoals gebruikt in deze taak staan bekend als 'algoritmes'. Een **algoritme** is een of formule voor het oplossen van een probleem.

Een recept voor het bakken van een cake is een algoritme. Als je het recept volgt dan zal je als resultaat een cake hebben. Alle wiskundige formules, zoals de formule voor de oppervlakte van een driehoek ( $A=B*H/2$ ) zijn algoritmes: zij bieden jou een methode voor het bepalen van een nieuwe waarde gebaseerd op bestaande waarden. Algoritmes zoals de 'perfecte kwadraat'-techniek hierboven, zijn belangrijke tools voor probleem oplossen. Door het leren van veelgebruikte computeralgoritmes zoals deze, zal je meer uit je programmeerervaring halen.

Hier is een recept voor het bakken van cake uit een pak ...

1. Bereid het cakebeslag volgens de aanwijzingen op het pak.
2. Bak de cake zoals aangegeven op het pak – in twee lagen.
3. Laat de cakes 10 minuten afkoelen in de vorm.
4. Haal de cakes uit de oven en zet ze op een bakrooster.
5. Laat ze helemaal afkoelen.
6. Klop de puddingmix met melk 2 minuten met een klopper.
7. Smeer dit onmiddellijk uit op de bovenkanten van de cakes.
8. Leg de twee lagen cake op elkaar.



9. Bestrijk het geheel met slagroom.
10. Eet smakelijk!

Net zoals in een computerprogramma, volgt de bakker de stappen van het begin naar het eind. Aan het eind is er een heerlijke cake om van te smullen. Als een computer de stappen in een programma (het algoritme) volgt, dan wordt het gewenste resultaat bereikt. Er bestaat zelfs een gebied binnen de computerwetenschappen dat zich bezighoudt met BEWIJZEN dat een algoritme het gewenste resultaat zal geven. Dat is vergelijkbaar met het bewijzen van wiskundige stellingen.

**Docenten Tip:** Dit is een voorbeeld van een computeralgoritme:

Om een getal af te ronden op het dichtstbijzijnde gehele getal gebruik je:

$$A = \text{int}(A+0,5)$$

Waarom werkt dit? Wanneer het decimale deel van A kleiner is dan 0,5, dan blijft het getal wanneer je er 0,5 bij optelt kleiner dan het opvolgende hogere gehele getal. In dat geval kapt de functie `int( )` het getal af bij de decimale komma, waardoor alleen het gehele deek overblijft.

Hier is een voorbeeld:

Stel dat geldt  $A=3,4$  dan  $A+0,5 = 3,9$  en  $\text{int}(3,9) = 3$  (rond af naar beneden)

Stel dat geldt  $A=3,7$  dan  $A+0,5 = 4,2$  en  $\text{int}(4,2) = 4$  (rond af naar boven)

Natuurlijk heeft de TI-84 ook een ingebouwde functie `round( )` in het menu MATH NUM!