

In deze eerste les van Unit 2, gebruik je variabelen in een programma en ontdek je het effect ervan in het document.

**Doelen:**

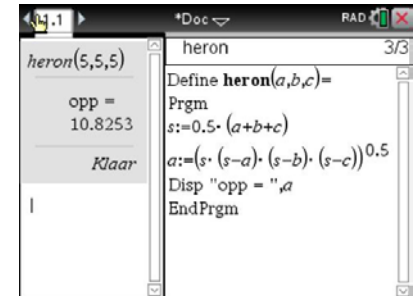
- Een variabele gebruiken in een programma
- De invloed leren kennen die het creëren van een variabele in een programma heeft op de rest van het document
- Verkennen wat de ‘speelruimte’ is van argumenten in een programma

**Een waarde opslaan in een variabele**

Soms is het nodig om een waarde toe te kennen aan een variabele binnen je programma. In ons voorbeeld gebruiken we de formule van Heron voor het berekenen van de oppervlakte van een driehoek op basis van de lengtes van de drie zijden. Dit is in een formule in twee stappen:

Bepaal eerst de halve omtrek:  $s = 1/2(a+b+c)$

En dan de oppervlakte:  $A = (s*(s-a)*(s-b)*(s-c))^{0,5}$



De programma-regels staan in de schermafbeelding rechts.

Merk op dat het symbool  $[:=]$ , en niet alleen een =-teken, is gebruikt om de waarde van een berekening toe te kennen aan een variabele.

$[:=]$  betekent “**krijgt (‘gets’) de waarde van ...**” of, eenvoudiger, “**krijgt**”, dus de opdracht

$s:=1/2*(a+b+c)$  wordt gelezen als “s **krijgt** de halve (1/2) omtrek van de driehoek”

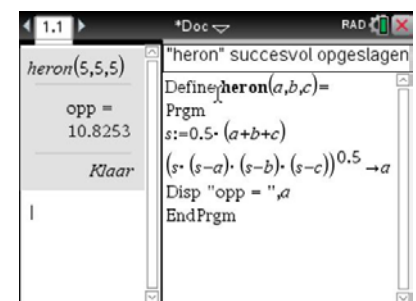
Op de TI-Nspire™ vind je deze combinatie van symbolen  $[:=]$  boven de ‘template’-toets [ctrl] [math].

Op een computer of op de rekenmachine kun je ook de twee tekens  $[:=]$  apart typen. Op de rekenmachine vind je de dubbele punt op de ‘interpunctietoets’ rechts van de letter G en het =-teken meteen onder [ctrl]-toets, dus op de rekenmachine is het eenvoudiger om gewoon [ctrl] [math] te typen.

**De operator ‘opslaan’ (store) gebruiken**

Je kunt de operator **opslaan** ( $\rightarrow$ ) gebruiken in plaats van **krijgt**  $[:=]$ .

Bekijk de programmaregels rechts. Merk op dat de opdrachten in de omgekeerde volgorde staan, waarbij eerst de berekening komt en dan de operator ‘store’ (**ctrl-var**) en dan de variabele. Dit is de volgorde waarin de opdrachten worden uitgevoerd (van links naar rechts).



Ofwel de operator ‘krijgt’  $[:=]$  of wel de operator ‘opslaan’ ( $\rightarrow$ ) kan worden gebruikt in elke toekenningso opdracht. Onthoud alleen wel wat de volgorde in de opdracht is voor elke methode.

Beide methodes kennen een waarde (meestal het resultaat van een berekening) toe aan een variabele en daarom worden dit toekenningso opdrachten genoemd.



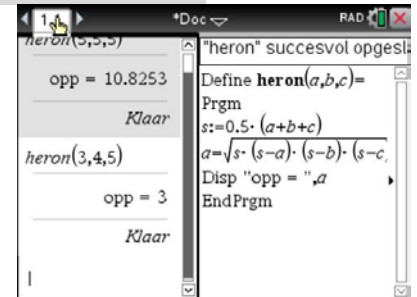
**Docenten Tip:** De operator **krijgt** ('gets')  $[\text{:=}]$  verwerkt de opdracht achterstevoren (van rechts naar links) omdat de uitdrukking aan de rechterkant wordt uitgevoerd en de waarde die het resultaat daarvan is wordt opgeslagen in de variabele aan de linkerkant van de operator. Dit is typerend voor de meeste programmeertalen.

De operator **opslaan** ('store') ( $\rightarrow$ ) is hetzelfde als die operator op de TI-84 Plus serie en de opdracht wordt uitgevoerd zoals die wordt gelezen (van links naar rechts).

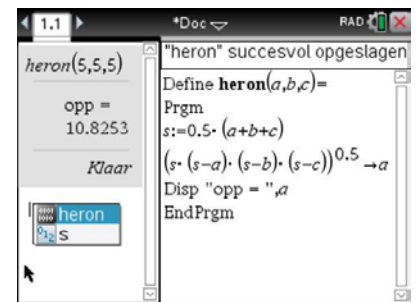
Laten we nu het volledige programma **heron(a,b,c)** zoals dat te zien is in de schermafbeelding invoeren.

Als het programma is ingevoerd controleer dan de syntax en sla het programma op door **menu > Syntax controleren en opslaan > Syntax controleren en opslaan** te selecteren (of de sneltoetscombinatie **ctrl-B** op de rekenmachine te gebruiken.)

Test het programma met de bekende waarden 3, 4, 5 om je ervan te verzekeren dat je het resultaat 6 krijgt voor de oppervlakte (*waarom is die 6?*)



Selecteer vervolgens in de toepassing *Rekenmachine (Calculator)* de **var**-toets en let op de lijst met variabelen. De variabelen **heron** en **s** verschijnen in de lijst. **heron** is de naam van het programma en **s** is gecreëerd door het programma. Maar **a**, **b**, en **c** zijn ook gebruikt in het programma. Waar zijn zij?



Het antwoord zit in het feit dat **a**, **b**, & **c** *argumenten* (parameters) zijn voor het programma en, als argumenten, alleen *binnen* het programma bestaan en niet gecreëerd zijn in het document. In feite willen we eigenlijk de variabele **s** ook niet in het document hebben! In de volgende les zullen we leren hoe we dit 'bij-effect' kunnen verhelpen.

Vergeet niet om het document op te slaan zodat je het programma opslaat! We gaan dit programma gebruiken in de volgende les.

**Docenten Tip:** De driehoek met zijden 3, 4, 5 is een rechthoekige driehoek en dus is de oppervlakte:  $0,5 \cdot 3 \cdot 4 = 6$ .

Onthoud dat een variabele is gedefinieerd in de huidige *opgave* en niet in het gehele document. Met het creëren van een nieuwe opgave in een document begin je met een hele nieuwe (lege) verzameling van variabelen; dus  $f_1(x)$  in opgave 1 is niet hetzelfde als  $f_1(x)$  in opgave 2.