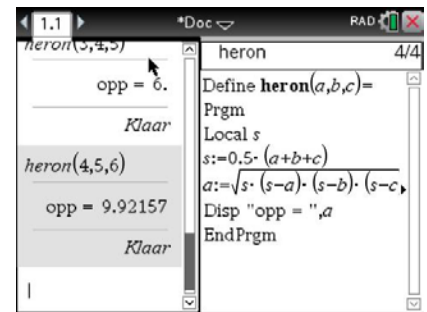


In deze derde les van Unit 2 leer je over de invloed van functies op globale variabelen.

Doelen:

- De invloed van een functie op globale variabelen vergelijken met die van programma's
- Programmacode kopiëren en plakken van het ene programma naar het andere of binnen hetzelfde programma

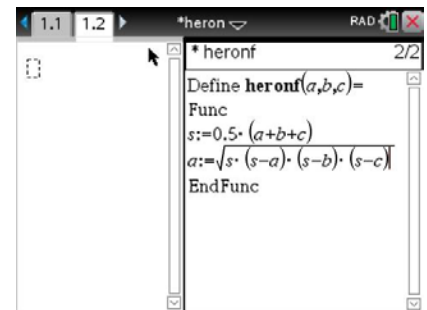
We beginnen deze les met het document dat het programma **heron** bevat wat je hier rechts ziet.



Voeg nog een *Rekenmachine*-toepassing toe aan het document en creëer dan een nieuw programma. Verander het Type in **Functie**.



Voeg de argumenten *a*, *b*, *c* toe binnen de haakjes en kopieer de twee toekenningsopdrachten van het programma naar de functie.



Om te *kopiëren* en *plakken* op de rekenmachine, doe je het volgende:

- Houd de **SHIFT**-toets ingedrukt terwijl je de tekstcursor over de gewenste tekst beweegt.
- Druk op **ctrl-C** om de tekst naar het klembord te kopiëren.
- Schakel over naar de functie en plaats de cursor op de gewenste positie
- Druk op **ctrl-V** om te plakken



Tech Tip: Op de TI-Nspire App voor de iPad tik je op de tekst en houd deze vast om hem te selecteren en trek je aan de uiteinden om alle gewenste te tekst te markeren. Selecteer *kopiëren* en tik dan op *plakken* en houd vast om dit te selecteren.

10 minuten programmeren

TI-NSPIRE TECHNOLOGY

Om het programma op te slaan selecteer **menu > Syntax controleren en opslaan > > Syntax controleren en opslaan** (of gebruik je de sneltoetscombinatie **ctrl-B** op de rekenmachine.)

Schakel over naar de toepassing Rekenmachine en voer de functie uit.

Zie je de foutmelding?

Wat is er mis?

De variabele **s** is een globale variabele omdat deze niet lokaal is 'gedeclareerd'. Dit is niet toegestaan in een functie, hoewel het dat wel is in een programma. Herinner je dat het doel van een functie is om een waarde terug te geven. Om globale variabelen te beschermen tegen onbedoelde veranderingen (de 'neveneffecten') verhindert het besturingssysteem dat functies globale variabelen veranderen.

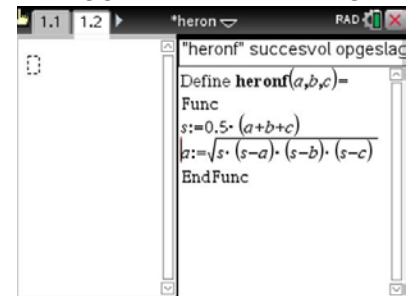
De oplossing: voeg de opdracht **Local s** toe aan het begin van de functie, sla het programma op met **menu > Syntax controleren en opslaan > > Syntax controleren en opslaan** (or door de sneltoetscombinatie **ctrl-B** op de rekenmachine te gebruiken) en test vervolgens de functie. Voila!

Docenten Tip: Merk op dat in de laatste stap last step de *Return*-opdracht ontbreekt in de functie. Zonder een *Return*-opdracht zal een functie de laatste waarde die is berekend teruggeven. Daarom wordt de waarde van *a* teruggegeven door de functie. Hoewel het niet per se nodig is moet je toch altijd een *Return*-opdracht gebruiken om duidelijk te zijn.

De *Return*-opdracht kan overal in een functie geplaatst worden maar wanneer deze eenmaal is uitgevoerd stopt de functie en worden er verder geen opdrachten meer verwerkt. Dit wordt belangrijk in volgende units waarin aftakkingen (if-opdrachten) en lussen voorkomen.

UNIT 2: OEFENBLAD 3

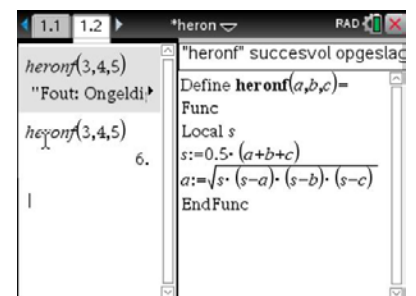
DOCENTENHANDLEIDING



```
heronf" succesvol opgesla  
Define heronf(a,b,c)=  
Func  
s:=0.5*(a+b+c)  
p:=sqrt(s*(s-a)*(s-b)*(s-c))  
EndFunc
```



```
heronf(3,4,5)  
"Fout Fout  
Ongeldig in een functie of de huidige uitdrukking  
Beeld Ga naar Annuleer
```



```
heronf(3,4,5)  
"Fout: Ongeldi  
heronf(3,4,5)  
6.  
Define heronf(a,b,c)=  
Func  
Local s  
s:=0.5*(a+b+c)  
a:=sqrt(s*(s-a)*(s-b)*(s-c))  
EndFunc
```