

Stroomschema's maken op papier



Een programma direct maken in *Scratch*, gaat vaak wel goed als het een klein programma is. Als het programma groter en moeilijker is, is het lastig om goed te zien welk commando waar moet komen. Je maakt dan sneller fouten.

Het is ook moeilijker om het programma dan netjes te maken. Een geordend programma werkt sneller en beter dan een programma dat rommelig is. Zo netjes mogelijk programmeren wordt *gestructureerd programmeren* genoemd. Een structuur is een ander woord voor hoe iets in elkaar zit.

Om je een handje te helpen bij het maken van een goed programma, gebruik je een *stroomschema*. Een stroomschema is een soort tekening. De tekening bestaat uit verschillende vakjes die met pijltjes met elkaar verbonden zijn. In de vakjes staan de commando's die een programma moet uitvoeren.

Met een stroomschema maak je eerst een oefenversie van een programma op papier, op de computer in een tekstverwerkingsprogramma, zoals *Word*, of een tekenprogramma zoals *Paint*. Omdat alles in losse vakjes staat, is duidelijk te zien hoe het programma werkt. Het stroomschema gebruik je daarna om stap voor stap je programma in *Scratch* te maken.

In dit aanvullende bestand wordt uitgelegd hoe je een stroomschema op papier maakt.


In dit hoofdstuk leer je:

- een stroomschema maken;
- een stroomschema starten en stoppen;
- pijlen toevoegen;
- een activiteit toevoegen;
- invoer en uitvoer toevoegen;
- een toelichting toevoegen;
- beslissingen toevoegen;
- lussen toevoegen;
- een procedure toevoegen.

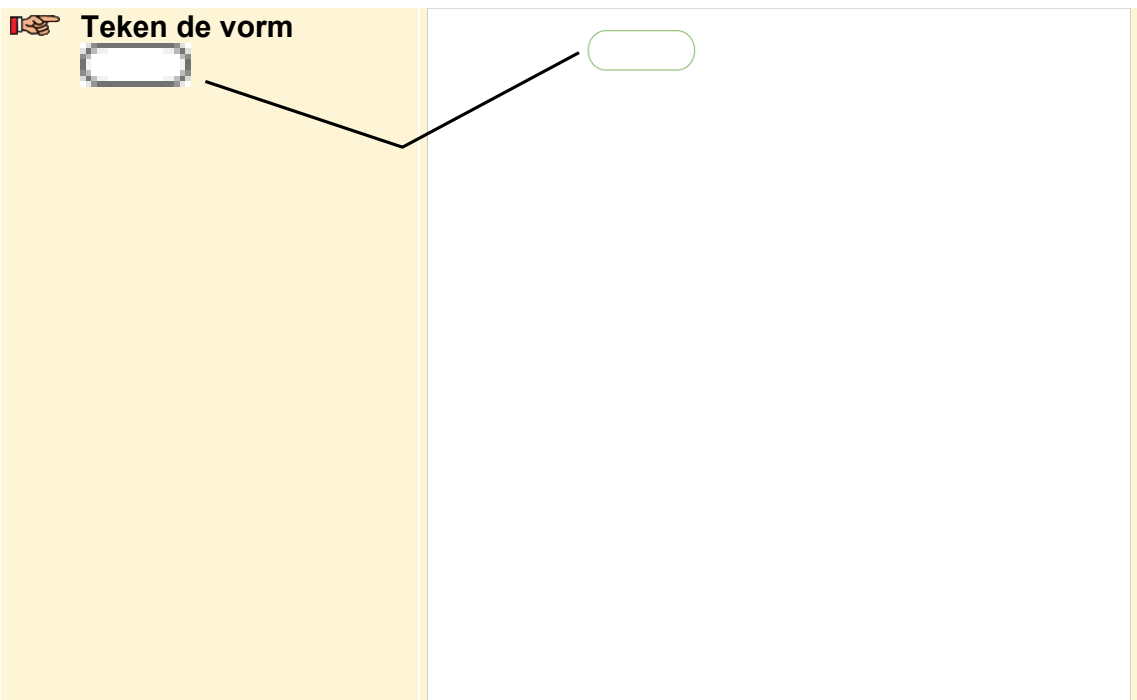
Een stroomschema starten

In dit voorbeeld wordt het stroomschema gemaakt op papier. Het is handig als dit een groot vel papier is, maar anders kan je ook verschillende velletjes aan elkaar plakken. Daarnaast heb je een potlood nodig. Een pen kan ook, maar een potlood is handiger als je iets wilt uitgummen.

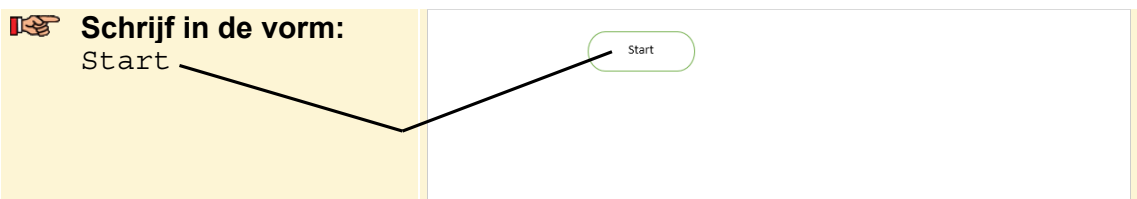
Pak een vel papier en een potlood

Een programma start altijd met een bepaald commando. Bijvoorbeeld het commando dat zegt dat je op de groene vlag moet klikken. Een stroomschema start ook altijd met een commando. Daar gebruik je deze *start*-vorm  voor. Met de start-vorm maak je duidelijk waar het stroomschema, en dus het programma, precies start.

Je voegt deze vorm toe aan je vel papier. Het is slim om dat tussen de kantlijn links en het midden van het document te doen. Je houdt dan genoeg ruimte links en rechts over voor de rest van het stroomschema. De meeste ruimte moet je rechts houden:

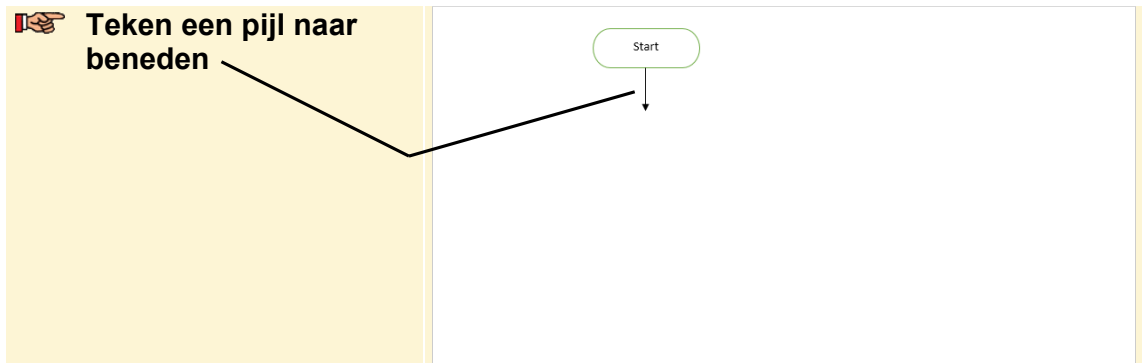


Je gaat ook nog een tekst in de vorm zetten. Daarmee vertel je wat er gebeurt in je programma. Je mag zelf de tekst verzinnen, als hij maar duidelijk genoeg is. In de start-vorm zet je meestal de tekst *Start*.



Een pijl toevoegen

Voordat je een volgende vorm in je stroomschema zet, teken je een pijl. Die pijl laat zien in welke volgorde het programma wordt uitgevoerd. Gewoonlijk is dat van boven naar beneden. Onder de pijl zet je dan straks de volgende vorm. Je zet een pijl altijd in het midden van een vorm:

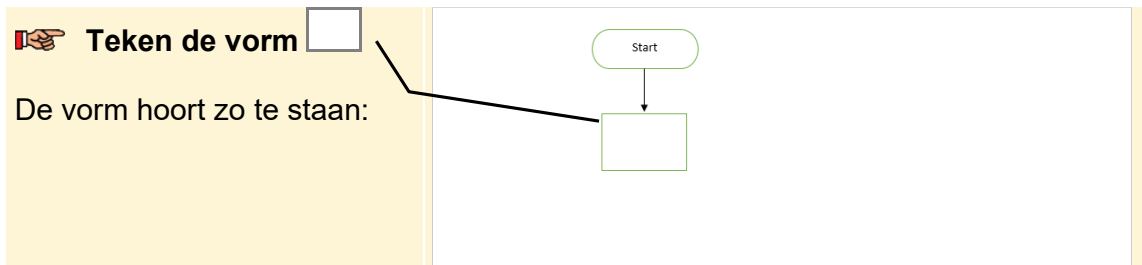


Een activiteit toevoegen

Als het begin van het stroomschema klaar is, voeg je de rest van het programma toe. Gewone commando's, zoals

ga naar x: 0 y: 0

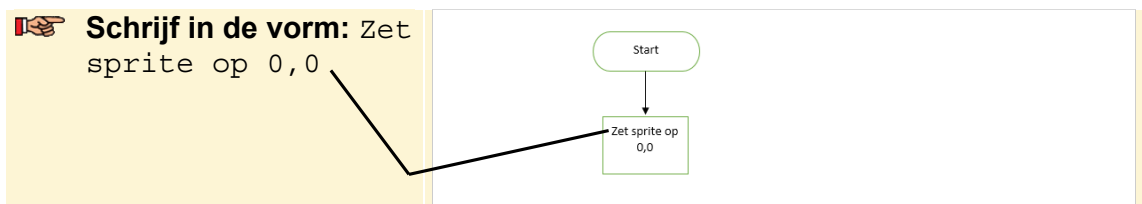
, voeg je toe met een *activiteit-vorm* . Die vorm zet je onder de pijl, recht onder de eerste vorm:



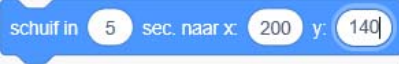
Je zet nu nog een tekst in de activiteit-vorm. Dit is het commando dat in het

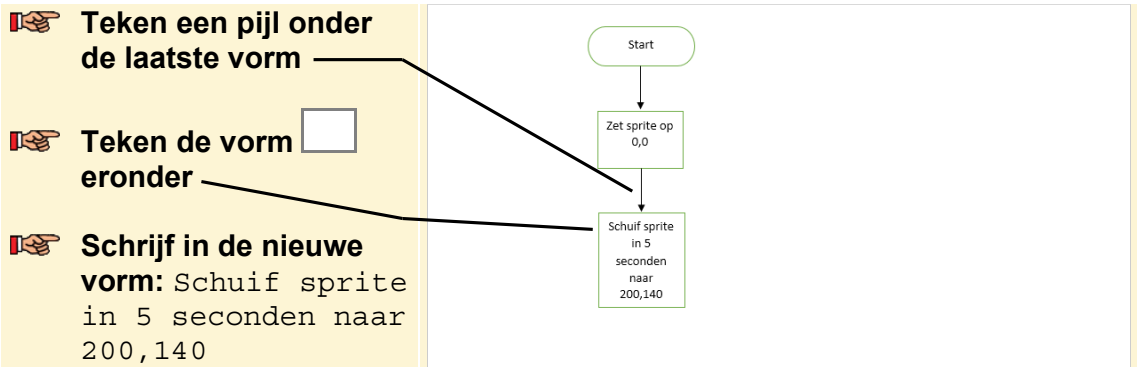
ga naar x: 0 y: 0

programma uitgevoerd wordt, bijvoorbeeld . Je hoeft niet precies de tekst van een commando in *Scratch* over te schrijven. Het gaat erom dat het duidelijk is wat er in het programma gebeurt:

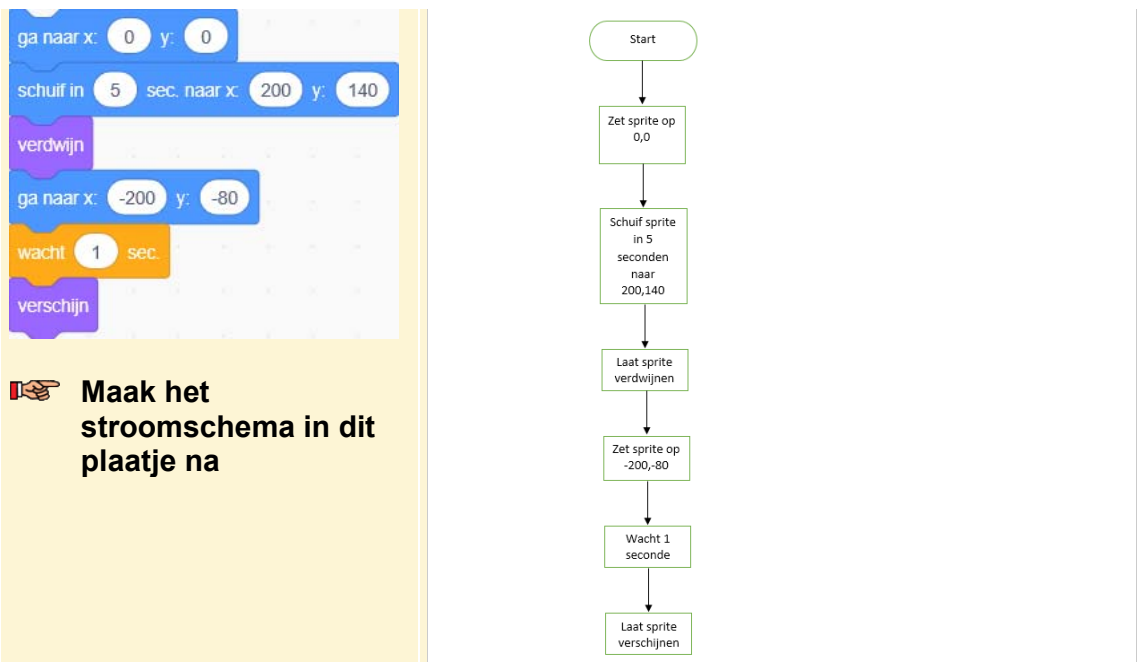


Je hebt een activiteit-vorm gemaakt.

Je kunt nog meer activiteit-vormen onder elkaar zetten in je stroomschema. Je voegt een activiteit-vorm toe voor het commando  :




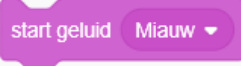
Op dezelfde manier maak je een heel stroomschema voor een eenvoudig programmadeel. Als je dat zou doen voor het programmadeel in *paragraaf 1.10 Een programma initialiseren* van het boek, ziet dat er zo uit:



Uitvoer toevoegen


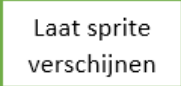
Invoer (input) en uitvoer (output) van gegevens zijn belangrijk in een programma. Bijvoorbeeld het typen van een getal of het laten zien van de uitkomst van een som.

Voor invoer en uitvoer gebruik je dezelfde *invoer/uitvoer-vorm* .


In het voorbeeld voeg je een uitvoer-vorm toe voor  en






in de *paragraaf 1.11 Tekst en geluid toevoegen* van het boek:

 **Teken een pijl omlaag onder** 

Je zet een tekst in de uitvoer-vorm. Dit is het commando



 **Teken de vorm** 


 **Schrijf in de vorm:**
Geluid Miauw






Je zet een uitvoer-vorm voor het commando



onder aan het stroomschema:


 **Teken een pijl omlaag**

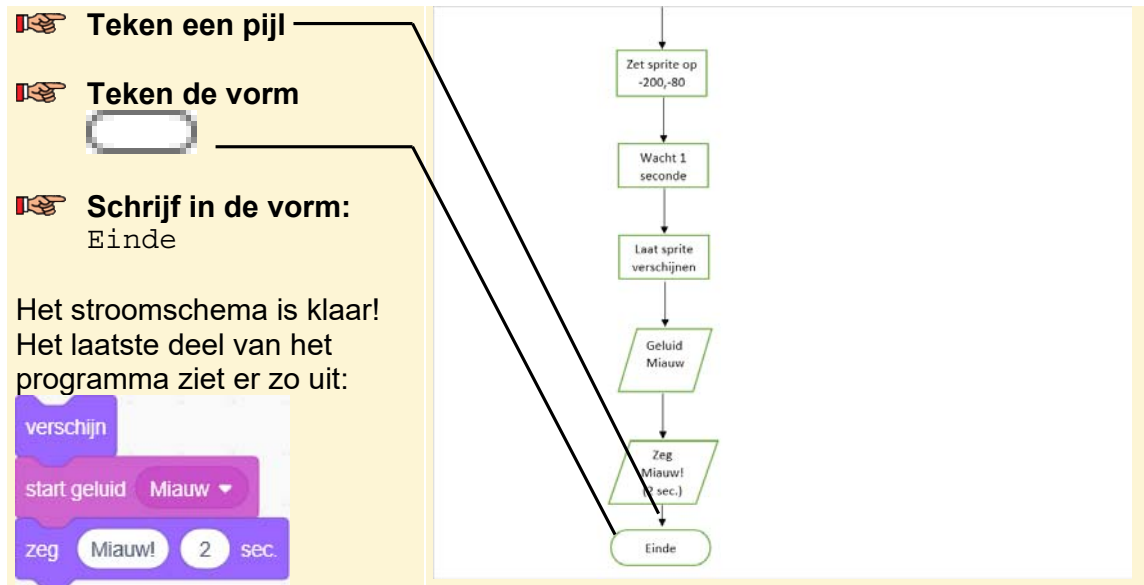
 **Teken de vorm** 

 **Schrijf in de vorm:** Zeg
Miauw! (2 sec)



Het einde toevoegen

Net zoals je laat zien waar het programma start, laat je ook zien waar het programma stopt. Daar gebruik je de *einde-vorm*  voor:




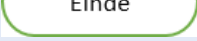
The image shows a Scratch interface with a flowchart on the right and code blocks on the left. The flowchart consists of the following steps: 'Zet sprite op -200,-80', 'Wacht 1 seconde', 'Laat sprite verschijnen', 'Geluid Miauw', 'Zeg Miauw! (2 sec.)', and 'Einde'. The 'Einde' block is highlighted with a red circle. On the left, there are three instructions with hand icons: 'Teken een pijl' pointing to the flowchart arrows, 'Teken de vorm' pointing to the 'Einde' block, and 'Schrijf in de vorm: Einde' pointing to the text 'Einde' in the code block. Below these instructions, there are three code blocks: 'verschijn', 'start geluid Miauw', and 'zeg Miauw! 2 sec.'. A text box says 'Het stroomschema is klaar! Het laatste deel van het programma ziet er zo uit:'.

Je kunt het stroomschema nu gebruiken om je programma mee te maken.

Tip


Een programma ergens anders stoppen


Je hoeft een programma niet altijd onderaan te stoppen. Je kunt het ook op een

andere plek stoppen met  in *Scratch*. Je zet  dan op die plek in het stroomschema.

Een toelichting toevoegen


Je maakt nog een ander stroomschema. Daarvoor kan je bijvoorbeeld op de achterkant van het papier verdergaan of je pakt een nieuw vel papier.

Soms is het handig om wat extra informatie bij een vorm in een stroomschema te zetten. Bijvoorbeeld omdat de tekst niet helemaal in de vorm past of omdat de tekst niet duidelijk genoeg is. Die extra informatie zet je in een *toelichting*. Daarvoor gebruik je de vorm .


In dit voorbeeld maak je een toelichting bij . Daarin zet je dat het programma start door op de groene vlag  te klikken met het commando




Je zet een toelichting rechts naast de vorm waar hij bij hoort:

Teken de vorm 

Schrijf in de vorm:
start

Teken de vorm rechts naast 


Schrijf in de vorm:
Klik op de groene vlag

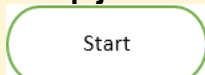
Teken een lijn van naar 


De toelichting is klaar.

Invoer toevoegen

Voor het invoeren van gegevens, gebruik je dezelfde vorm als voor uitvoer: . In

het voorbeeld voeg je een invoer-vorm toe voor  in de *paragraaf 3.1 Werken met het als/dan-commando* in het boek:


Teken een pijl omlaag onder 



Teken de vorm onder de pijl onder 

Je zet een tekst in de invoer-vorm:

Typ: Wat is de temperatuur?



Een beslissingscommando toevoegen


Vaak gebruik je in een programma beslissingscommando's of als/dan-commando's. Om die in je stroomschema te zetten, gebruik je de *beslissing-vorm* .

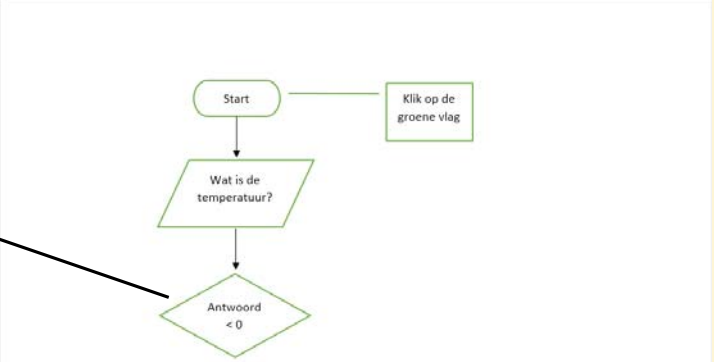
 **Voeg een pijl omlaag toe onder** 

Je zet de vorm onder aan het stroomschema en zet er een tekst in. Dat is de vergelijking waar de beslissing aan moet voldoen. In dit voorbeeld is dat het


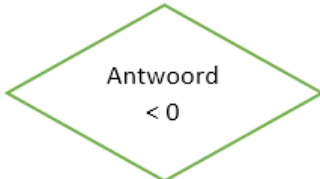
commando  :

 **Teken de vorm** 



 **Schrijf in de vorm:**
Antwoord < 0

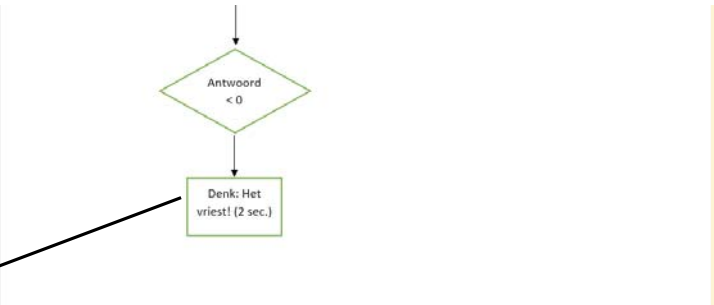


Tot nu toe voegde je bij een vorm een pijl toe. Bij een beslissing-vorm voeg je twee pijlen toe. Een voor als de vergelijking klopt. Die loopt recht naar beneden:

 **Teken een pijl omlaag onder** 

Hieronder zet je de commando's die worden uitgevoerd als de vergelijking klopt. In dit voorbeeld is dat  :

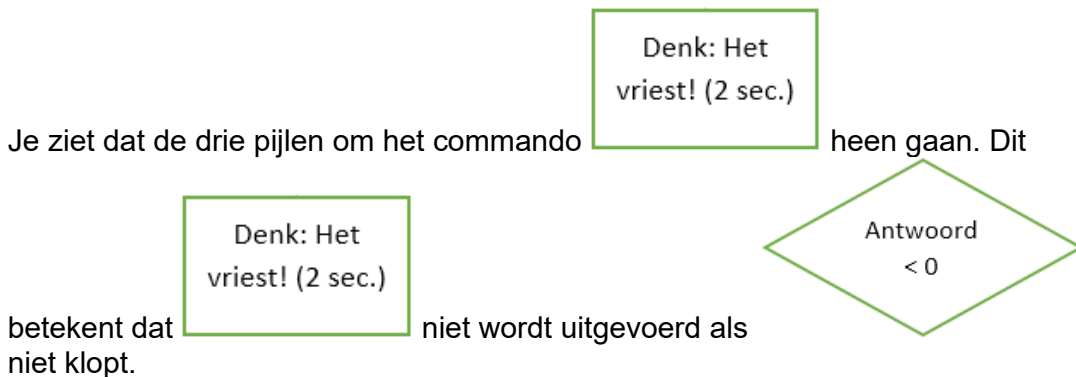
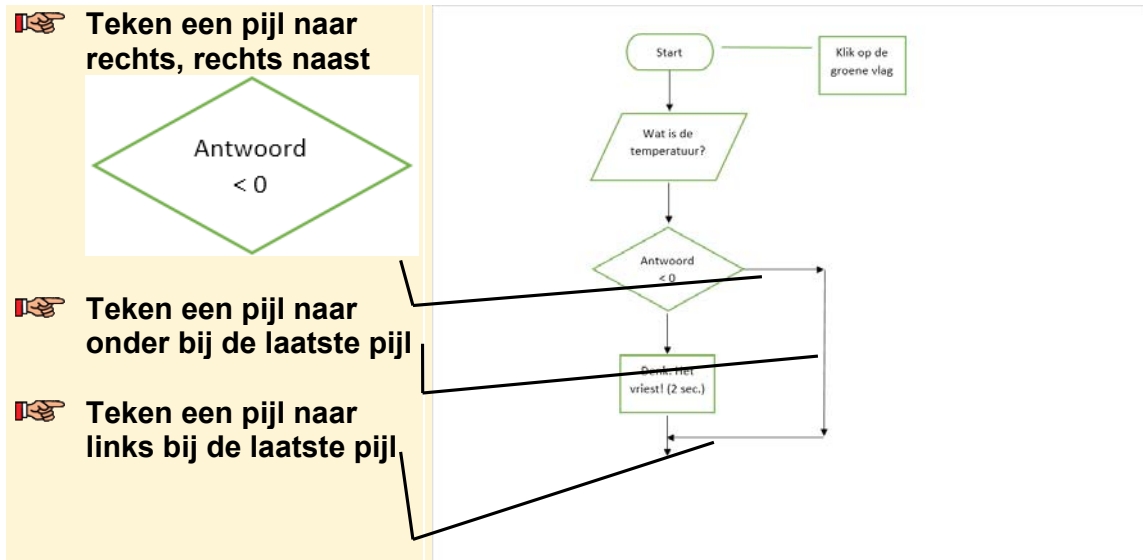
 **Teken een activiteit-vorm voor**

zoals in het plaatje



Teken een pijl omlaag onder

Denk: Het vriest! (2 sec.)

De andere pijl zet je aan de rechterkant van de beslissing-vorm. Deze geeft de route aan als de vergelijking niet klopt. In dit voorbeeld slaat de pijl het commando over. Hiervoor gebruik je een aantal pijlen:

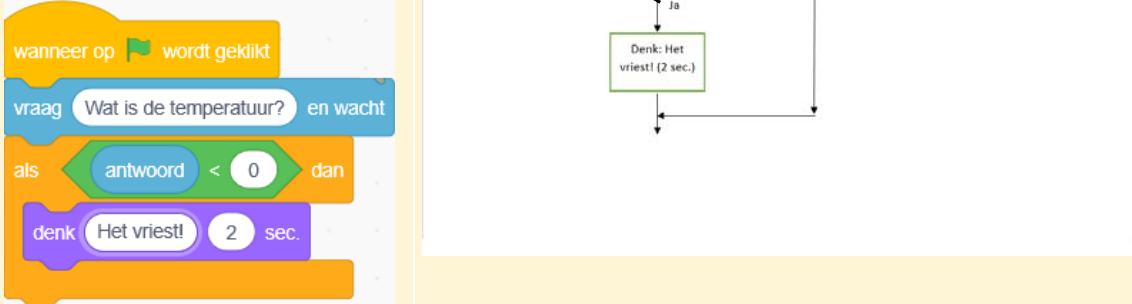


Je geeft nu nog bij de beslissing-vorm aan wanneer de pijl omlaag en wanneer de pijl naar rechts wordt genomen. Daarom zet je bij de route omlaag *Ja* en de route naar rechts *Nee*.

Schrijf bij de pijl naar rechts: Nee

Schrijf bij de pijl naar beneden: Ja

De beslissing-vorm en alles wat erbij hoort is klaar:




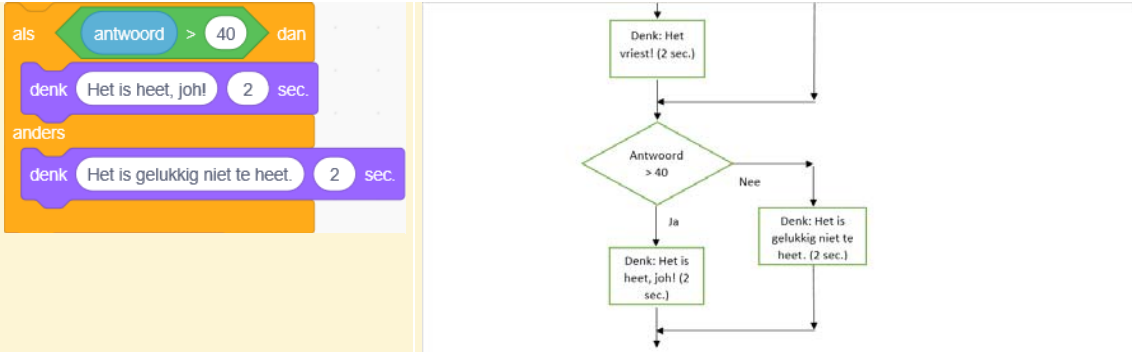
The image shows a Scratch code editor on the left and a flowchart on the right. The code blocks include: 'wanneer op groene vlag wordt geklikt', 'vraag 'Wat is de temperatuur?' en wacht', 'als antwoord < 0 dan', and 'denk 'Het vriest!' (2 sec.)'. The flowchart starts with 'Start', followed by 'Klik op de groene vlag', then a question box 'Wat is de temperatuur?'. It leads to a decision diamond 'Antwoord < 0'. The 'Ja' path leads to 'Denk: Het vriest! (2 sec.)' and then loops back to the question box. The 'Nee' path leads to the right and loops back to the question box.

De volgende commando's zet je gewoon onder de laatste pijl.



Een stroomschema maken voor een  commando werkt bijna

hetzelfde als voor . Het enige verschil is dat je ook bij de route voor Nee commando's neerzet. Dat ziet er dan bijvoorbeeld zo uit:




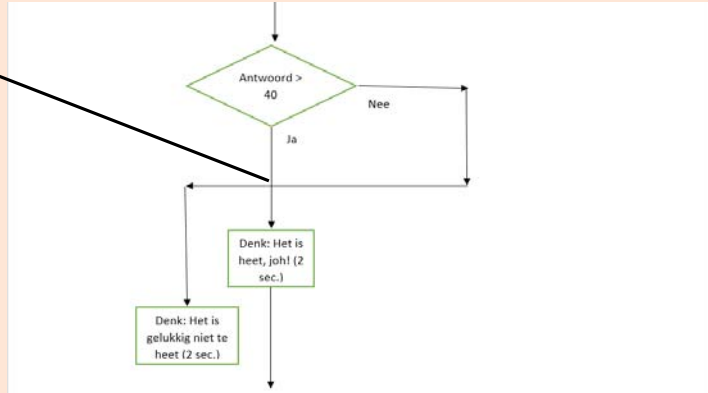
The image shows a Scratch code editor on the left and a flowchart on the right. The code blocks include: 'als antwoord > 40 dan', 'denk 'Het is heet, joh!' (2 sec.)', 'anders', and 'denk 'Het is gelukkig niet te heet.' (2 sec.)'. The flowchart starts with 'Denk: Het vriest! (2 sec.)', followed by a decision diamond 'Antwoord > 40'. The 'Ja' path leads to 'Denk: Het is heet, joh! (2 sec.)'. The 'Nee' path leads to 'Denk: Het is gelukkig niet te heet. (2 sec.)'. Both paths then loop back to the decision diamond.

Let op!

Als je goed en netjes wilt programmeren, zorg je er voor dat de pijlen in je stroomschema elkaar niet kruisen:

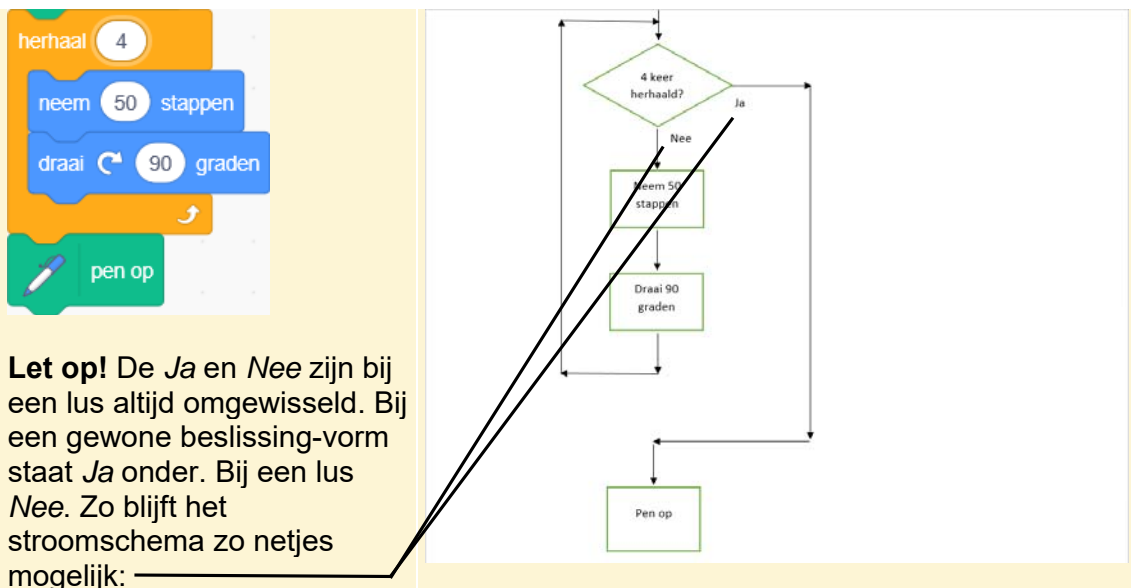
Als de pijlen elkaar wel kruisen:

 **Verander je stroomschema, zodat de pijlen wel goed lopen**



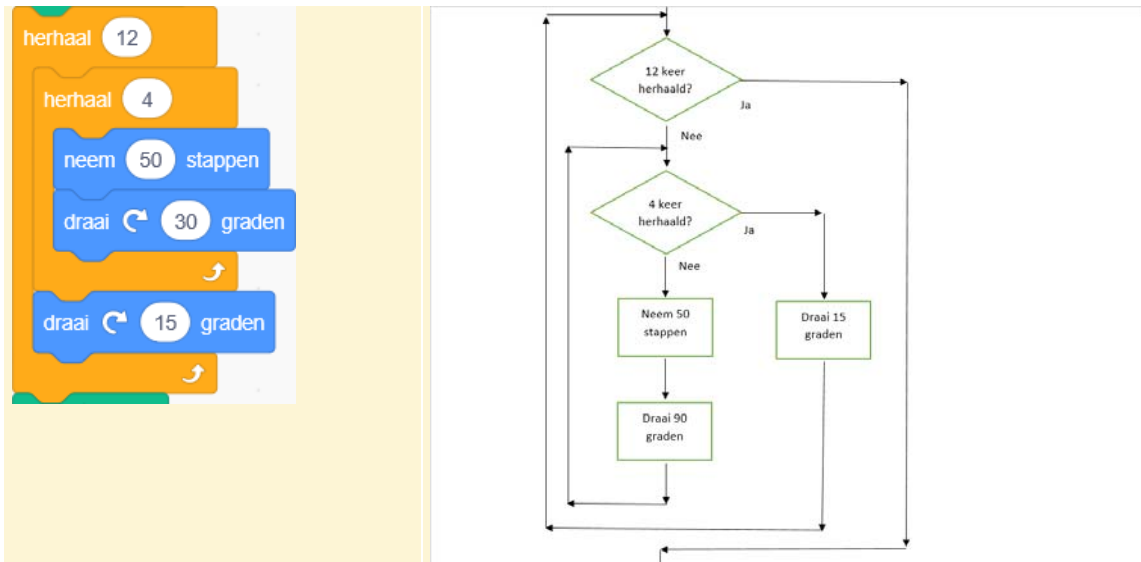
Een lus toevoegen

Een lus is eigenlijk een beslissing-vorm die ieder keer bij zichzelf terugkomt. Dat gaat net zo lang door totdat de vergelijking klopt. Daarna ga je in het stroomschema verder met het volgende commando. Dat ziet er bijvoorbeeld zo uit in *paragraaf 4.4 De pen gebruiken* in het boek:

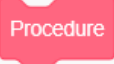



Let op! De *Ja* en *Nee* zijn bij een lus altijd omgewisseld. Bij een gewone beslissing-vorm staat *Ja* onder. Bij een lus *Nee*. Zo blijft het stroomschema zo netjes mogelijk:

Je kunt ook een lus nesten, zoals in *paragraaf 4.5 Geneste lussen* in het boek:





Een procedure toevoegen

Je kunt delen van een programma in een procedure zetten. In *Scratch* start je de procedure met het commando . In een stroomschema gebruik je daarvoor de *procedure*-vorm . Je zet bijvoorbeeld het initialisatiedeel van een programma in een procedure:

 **Kijk of je nog ruimte hebt op je papier of pak een nieuw vel papier**

 **Teken een  vorm**

 **Teken een pijl omlaag onder **

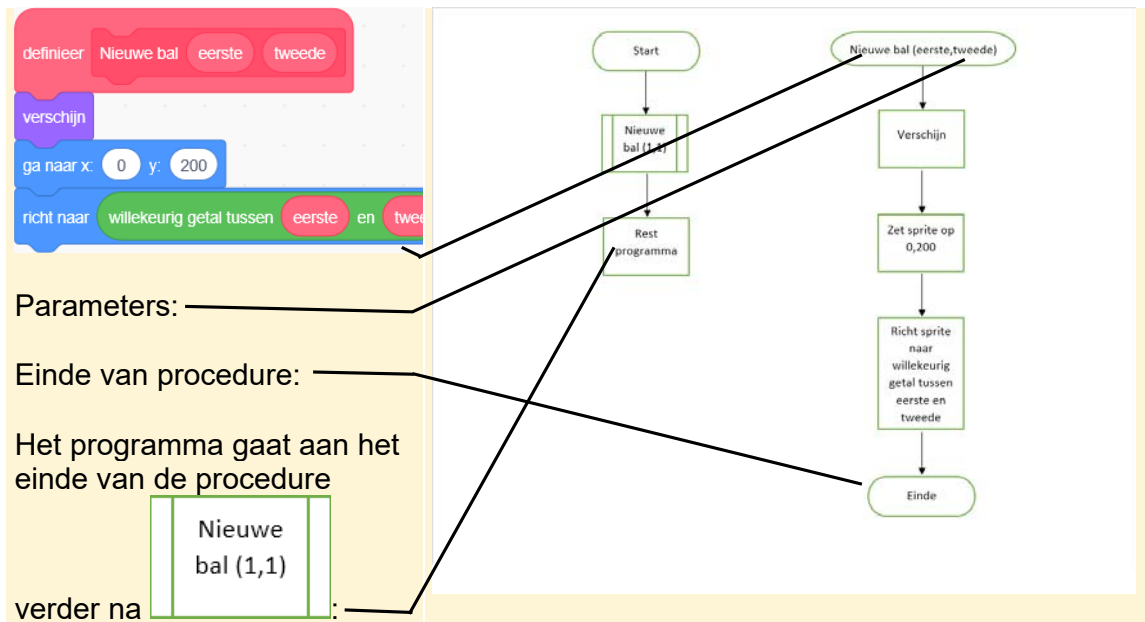
Je noemt de procedure **Nieuwe bal** zoals in *paragraaf 6.8 Een procedure gebruiken*:



Voor de procedure zelf gebruik je de start-vorm . Alleen zet je daar geen *Start* in maar de naam van de procedure. In dit voorbeeld is dat *Nieuwe bal*. De rest van de procedure is een deel van het programma en werkt dan ook met dezelfde

vormen. Het einde geef je aan met .

De procedure zet je op een plek in het document waar nog lege ruimte is. Dat kan naast het programma zijn of op een andere pagina. In het voorbeeld van *paragraaf 6.8 Een procedure gebruiken* in het boek ziet dat er zo uit:



Je hebt in deze handleiding geleerd hoe je stroomschema's maakt. Je kunt nu voortaan eerst een stroomschema maken voordat je echt gaat programmeren in *Scratch*. Veel succes!