

# GEAVANCEERDE EV3 PROGRAMMEERLES

## Lijnvolgers: van basis tot proportioneel



Door Droids Robotics



# leerdoelen

1. Eigenschappen van verschillende lijnvolgers
2. Leer met het concept “evenredig” een proportionele lijnvolger te maken

Vereiste voorkennis: Basis lijnvolger, Lijnvolger op kleur, Kleuren sensor kalibreren, proportionele controle, Wiskundige blokken, Data Wires

# Welk programma werkt het best bij welke situatie?

## Simpele lijnvolger

- De meest basis-lijnvolger
- Schud veel bij scherpe bochten
- Goed voor beginnende teams → je leert over herhalingen en schakelopties

## 3 – stadia volger

- Het beste voor rechte lijnen
- Droids bevelen dit niet aan. Leer gewoon de evenredige lijnvolger. Je moet de gevestigde schakelaars kennen.

## Soepele lijnvolger

- Bijna gelijk aan basis
- Draait minder scherp
- Heeft moeite met scherpe bochten
- Goed voor beginnende teams → je leert over herhalingen en schakelopties

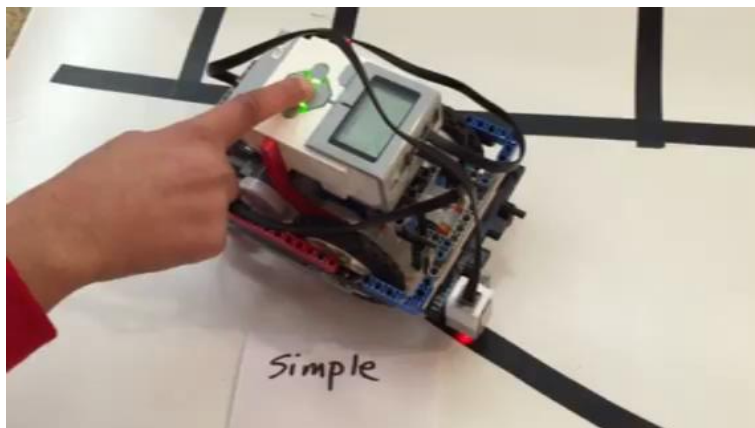
## proportionele volger

- Gebruikt de “P” van PID
- Maakt evenredige bochten
- Werkt goed op bochten en rechte lijnen
- Goed voor gevorderde en geavanceerde teams → Je moet wiskundige blokken en Data wires kennen.

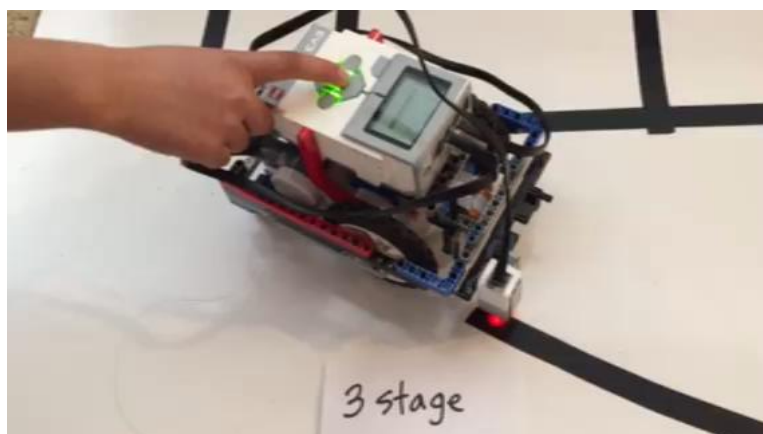
**Bekijk de filmpjes op de volgende 2 slides om alle lijnvolgers in werking te zien.**

# ronde lijn: Bekijk filmpjes

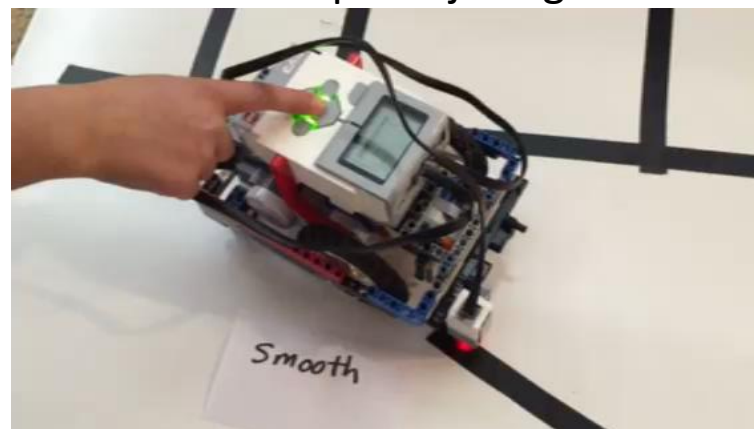
Simpele lijnvolger



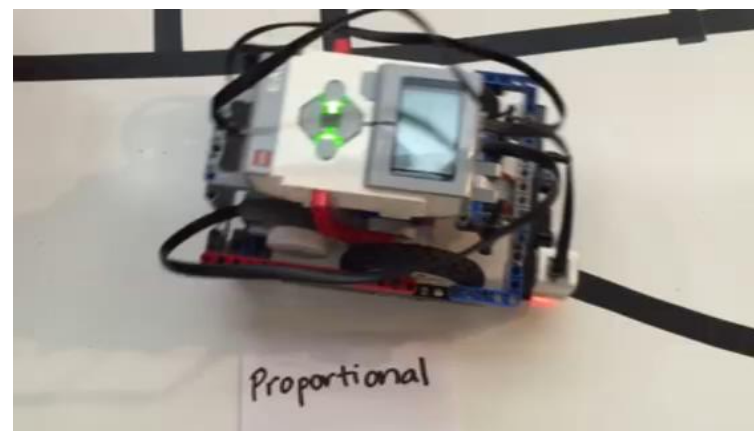
3-stadia lijnvolger



Soepele lijnvolger

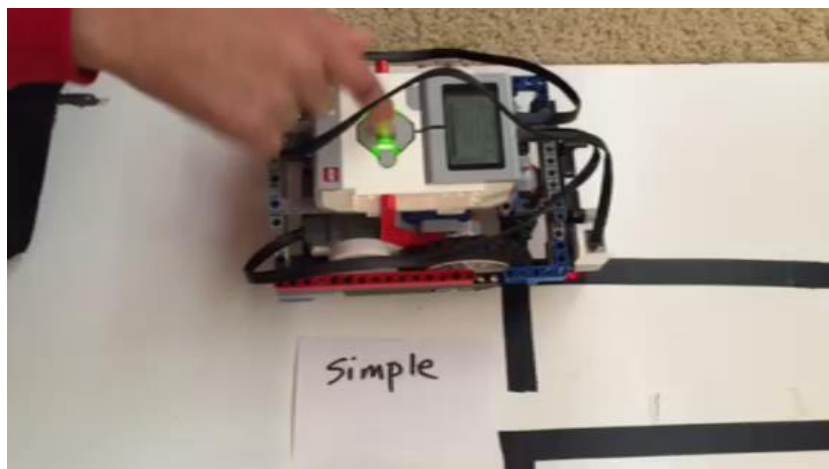


proportionele lijnvolger

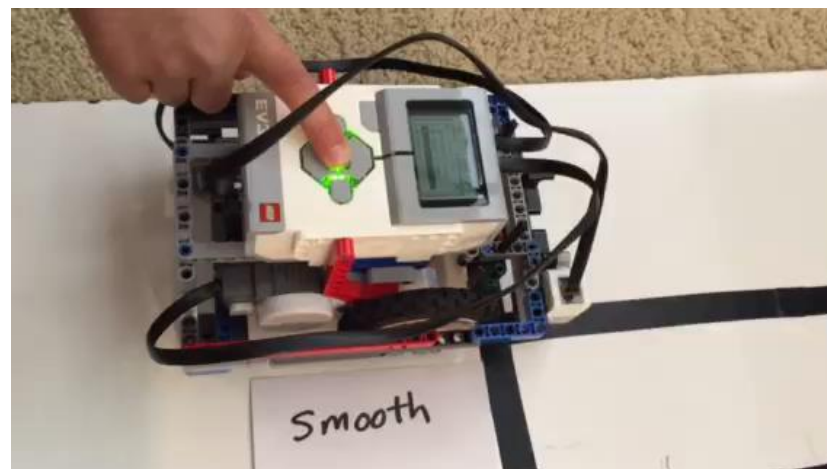


# Rechte lijn: Bekijk filmpjes

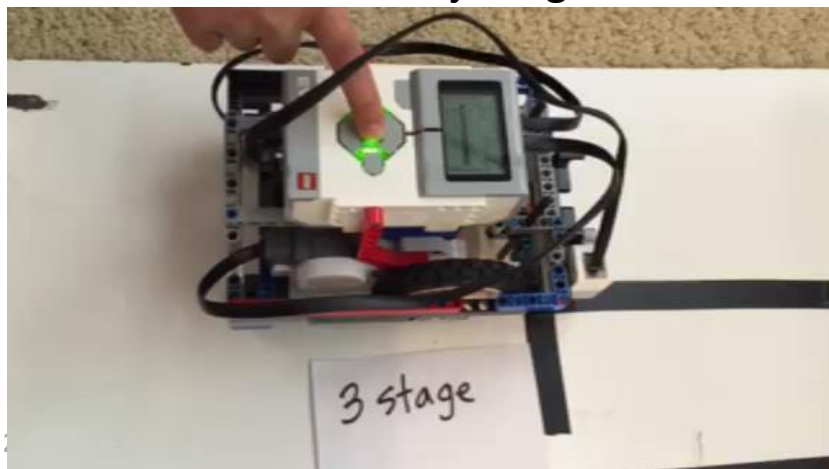
Simpele lijnvolger



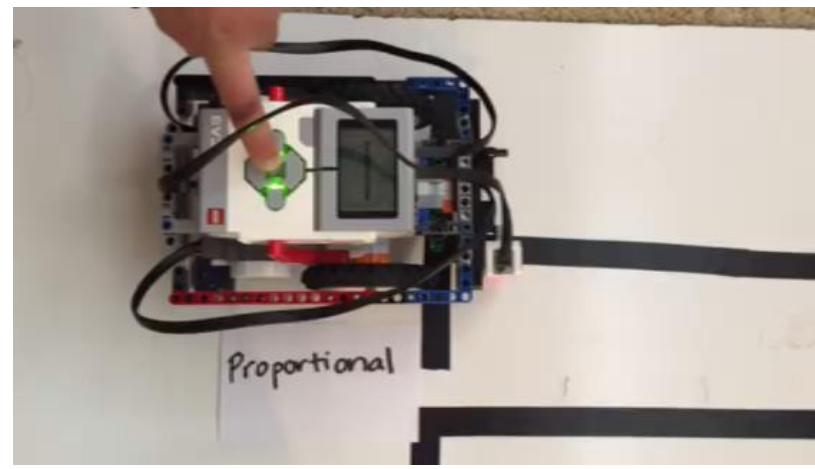
Soepele lijnvolger



3-stadia lijnvolger



proportionele lijnvolger



# 3 lijnvolger uitdagingen

- ➔ **Uitdaging 1:** Kun je een **simpele lijnvolger** schrijven? Hint: Bekijk Beginner: de basis lijnvolger les
- ➔ **Uitdaging 2:** Kun je een **soepele lijnvolger** schrijven? Hint: Verander hoe scherp de bochten worden bij een simpele lijnvolger.
- ➔ **Uitdaging 3:** Kun je een **3-stadia lijnvolger** schrijven waar de robot 3 opties heeft (links, rechts of rechtdoor) gebaseerd op de lezingen van de kleurensensoren?

# Nog iets over onze oplossingen

## ➤ KALIBRATIE:

- De programma's gebruiken de EV3 kleurensensor als lichtsensor.
- Je moet je sensoren kalibreren.
- Voor kalibreren gebruik de kleurensensor kalibratie les bij gevorderde lessen.

## ➤ POORTEN:

- De kleurensensor is verbonden met poort 3.
- Verander dit voor jouw robot.

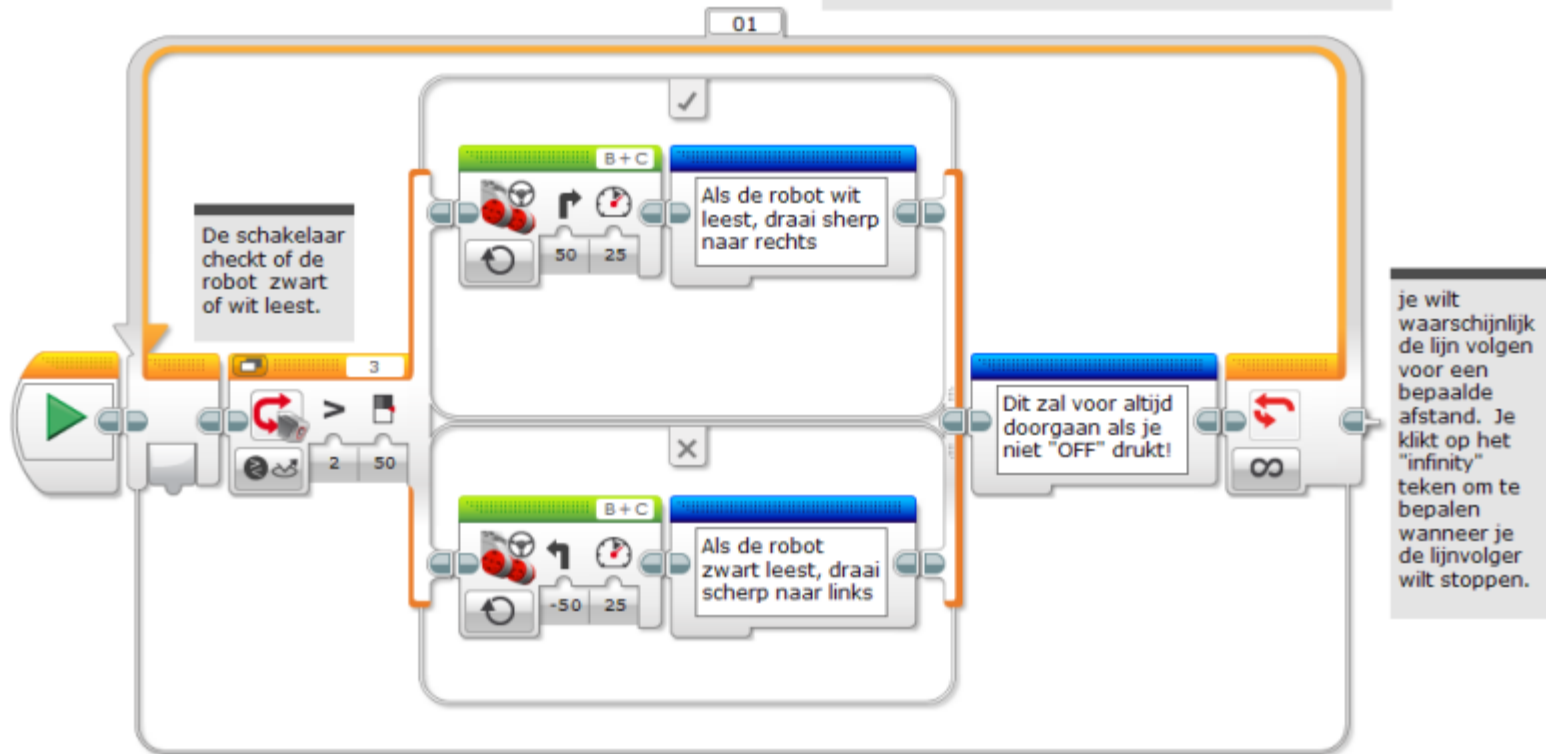
## ➤ WELKE KANT VAN DE LIJN:

- Kijk naar de code om te zien voor welke kant van de lijn deze is geschreven.

# Oplossing 1: Simpele lijnvolgger

Simpele lijnvolgger: het doel van dit programma is om een zeer simpel lijnvolg programma te maken die de linkerkant van een lijn volgt. Dit is het meest geleerde programma.

Let op: Dit programma gebruikt de kleurnesensoren in licht mode. Dit betekent dat je sensoren moet kalibreren. Lees onze kalibratie les voor dat je gaat beginnen!

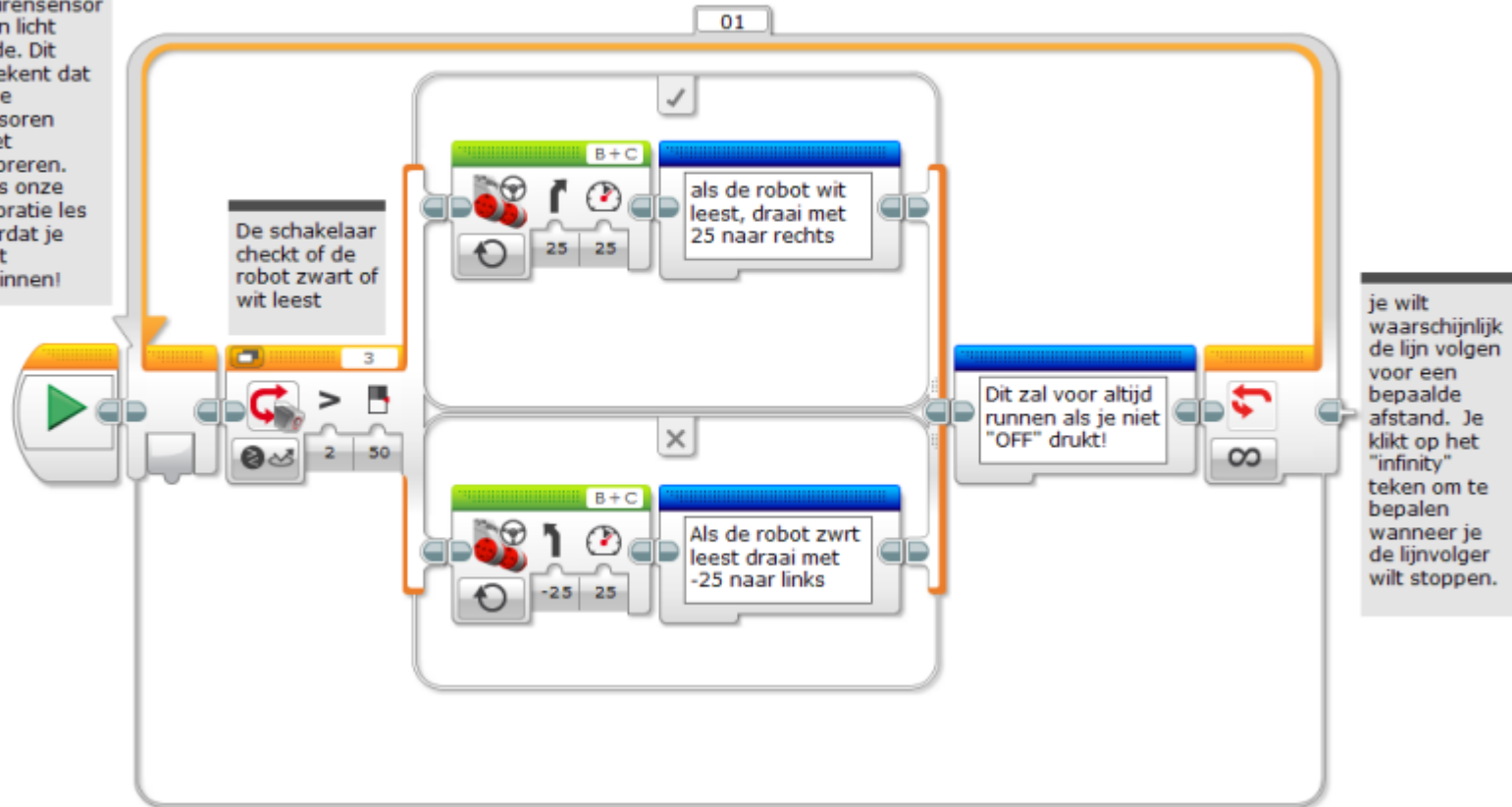




# Oplossing 2: Soepele lijnvolger

let op: dit programma gebruikt de kleurensensor en in licht mode. Dit betekent dat je de sensoren moet kalibreren. Lees onze kalibratie les voordat je gaat beginnen!

Soepele lijnvolger: Het doel van dit programma is om een simpele lijnvolger te maken, maar dan soepeler dan de eerste. Dit programma zal soepeler zijn omdat het minder snelle bochten maakt. Het enige verschil tussen de simpelere en de soepelere is dat het minder snel boschten maakt.



# Oplossing 3: 3-stadia lijnvolger

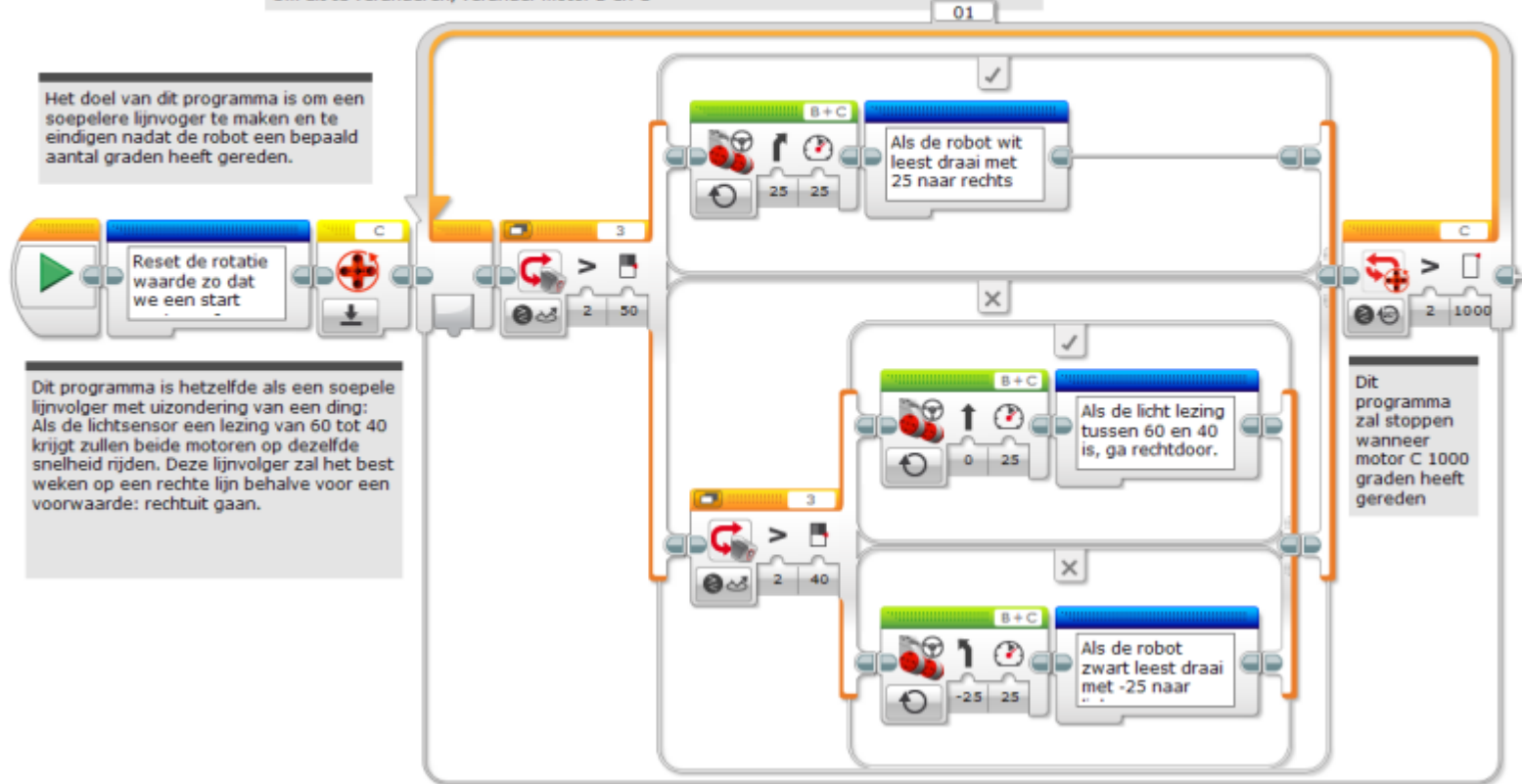
Let op: We presenteren deze lijnvolger omdat veel teams horen over deze en willen weten hoe je er een kan schrijven. Ons team beveelt deze lijnvolger niet aan maar juist de evenredige!

let op: dit programma gebruikt de kleursensoren in licht mode. Dit betekent dat je de sensoren moet kalibreren. Lees onze kalibratie les voordat je gaat beginnen!

Let op 1: Als B je rechter motor is, dan zal dit programma de linker kant van de lijn volgen.  
Let op 2: Als B je linker motor is, dan zal dit programma de rechter kant van de lijn volgen.  
Om dit te veranderen, verander motor B en C

Het doel van dit programma is om een soepele lijnvolger te maken en te eindigen nadat de robot een bepaald aantal graden heeft gereden.

Dit programma is hetzelfde als een soepele lijnvolger met uitzondering van een ding: Als de lichtsensoren een lezing van 60 tot 40 krijgt zullen beide motoren op dezelfde snelheid rijden. Deze lijnvolger zal het best weken op een rechte lijn behalve voor een voorwaarde: rechtuit gaan.



Dit programma zal stoppen wanneer motor C 1000 graden heeft gereden

# Uitdaging 4: Evenredige lijnvolger

**Uitdaging 4:** Kun je een **proportionele lijnvolger** schrijven dat de hoek van de bocht verandert op basis van hoe ver de robot van de lijn is?

Pseudocode:

1. Reset de rotatie sensor (Alleen nodig voor lijnvolgen voor een totale afstand)
2. Bereken de fout = Afstand tot de lijn = (Lichtsensoren lezen – doelwit lezen)
3. Vergroot of verklein de fout tot de correctie waarde. Pas je vergrotingsfactor aan om je robot soepeler te laten gaan.
4. Gebruik de correctieve-waarde (berekent in stap 3) om de robots bocht naar de lijn aan te passen.

# Oplossing: proportionele lijnvolger

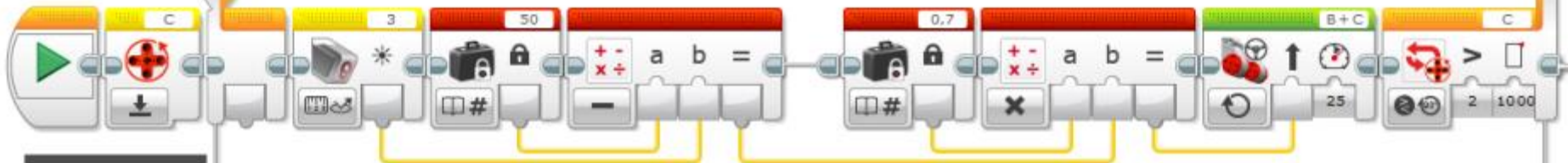
Let op: Dit programma gebruikt de klemesensoren in licht mode. Dit betekent dat je sensoren moet kalibreren. Lees onze kalibratie les voor dat je gaat beginnen!

Wij bevelen je team een evenredige lijnvolger te gebruiken zoals deze. Het zal de meest soepele lijnvolger van de 4 zijn. Er zijn betere lijnvolgers (die PID controle gebruiken), maar een lijnvolger die "P" gebruikt is een goed begin.

Een evenredige lijnvolger verandert de hoek van de bocht gebaseerd op hoever de lijn van de robot is.

01

Elke evenredige lijnvolger moet 2 delen hebben: Deel 1 berekent de error (In dit geval, hoe ver je bent van de lijn) en deel 2 berekent de correctie evenredig tot de error (in dit geval hoeveel graden de bocht). Je kan evenredige besturing ook met andere sensoren gebruiken. Het werkt heel goed!



Let op: Je hoeft niet de constante blok met een data-draad vast te maken. We deden dat omdat het meer vanzelfsprekend was dat we vermenigvuldigde met een constante van onze eigen keus.

**Deel 1: Bereken de error**

- Ons doel is om op de rand van de lijn te zijn (lichtsensor = 50). Het wiskundige blok hierboven, hoever onze robot is van ons doel van 50.
- De constante blok hierboven is ons doel. Je kunt het voor verschillende lijnen veranderen.
- Let op dat in het ergste geval, je licht sensor 0 of 100 leest (Ver van de lijn!). Dit geeft een error van 50 or -50.

**Deel 2: Bereken en gebruik error**

- We vermenigvuldigen de error van deel 1 met 0.7 om de bocht waarde te berekenen.
- We kozen 0.7 zodat wanneer we in ergste geval de lezing van 50 or -50 hebben, de besturing in de richtings blok hierboven 35 or -35 zal zijn, wat een scherpe bocht zou zijn.
- Je kan deze waarde veranderen om je lijnvolger op je behoefte te veranderen.

Deze lijn volger eindigt na 1000 graden. verander naar jou behoefte.

# Tips

Je krijgt betere resultaten als:

- ....je kleurensensor dichterbij de grond is (maar niet erop)
- ....je kleurensensoren beschermt tegen ander licht.
- ....je onthoudt om te kalibreren.

# Discussie Gids

## Simpele lijnvolger

+  
+  
-  
-

## Soepele lijnvolger

+  
+  
-  
-

## 3-stadia lijnvolger

+  
+  
-  
-

## Proportionele lijnvolger

+  
+  
-  
-

Vul de bovenstaande in bij de positieve en negatieve punten. Overweeg welke lijnvolger het beste is voor rechte lijnen of bochten. Overweeg of de robot veel zal schudden ja of nee.

# Credits

- This tutorial was created by Sanjay Seshan and Arvind Seshan from Droids Robotics.
  - Author's Email: [team@droidsrobotics.org](mailto:team@droidsrobotics.org)
- More lessons at [www.ev3lessons.com](http://www.ev3lessons.com)
- Translated by Sebastiaan Berting from Maerlant Robotica



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).