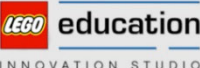


SPIKE SBC Teller programma: basis

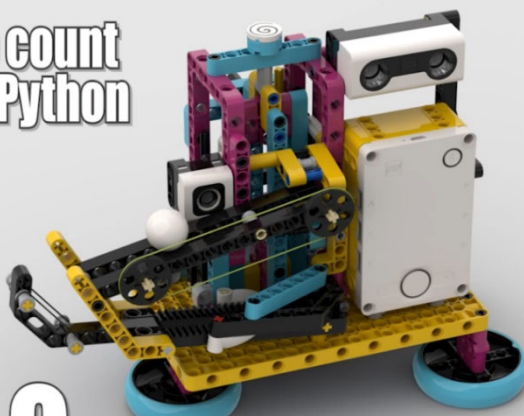




Learn how to count
in Scratch & Python

**SPIKE
Prime**

Small **B**all **C**ontraption by Dimi



Part 1:
SPIKE Prime
Scratch
coding

Part 2:
SPIKE Prime
Python coding

 **LEGO® Education Academy**
Certified Trainer

LEGO Education SPIKE - 1.1.4
Bestand Bekijken Venster Help

Variabelen

Maak een variabele

MOTOREN

BEWEGING

LIJCHT

Mijn blokken

Maak een blok

SBC Basis

B
D
F

wanneer programma wordt gestart

maak Teller 0

aanzetten

F licht omhoog

wanneer dichterbij dan 8 cm

D stelt snelheid in op 15 %

D start motor

wanneer kleur is

speel pieptoon af 108 gedurende 0.2 seconden

verander Teller met 1

Teller schrijven

Als het programma opstart, wordt deze code (stack) uitgevoerd.

Optioneel: Tellerwaarde op 0 zetten. (Dit hoeft eigenlijk niet, maar kan een goede gewoonte zijn voor later in andere programmeertalen.)

Pijl op scherm tonen die wijst naar de Afstandssensor.

"Ogen" van de Afstandssensor laten oplichten.

Telkens de Afstandssensor iets waarneemt dichterbij dan 8 cm, dan wordt deze code (stack) uitgevoerd.

Snelheid van motor D instellen op 15%

Motor D starten (deze motor zal blijven draaien tot we verder in het programma deze motor terug stoppen).

Telkens de Kleursensor iets wits ziet, dan wordt deze code (stack) uitgevoerd.

De SPIKE Hub gedurende 0,2 sec een hoge pieptoon laten afspelen.

Hier laten we de waarde van de variabele met 1 verhogen. (= 1 erbij tellen.)

De waarde van onze Teller variabele op het scherm van de Hub plaatsen.

SPIKE SBC Teller programma: Uitbreiding

LEGO Education SPIKE - 1.1.4

Bestand Bekijken Venster Help

Variabelen

Maak een variabele

Teller

maak Teller 0

verander Teller met 1

Maak een lijst

Mijn blokken

Maak een blok

MOTOREN

BEWEGING

LICHT

GELUID

GEBEURT...

BEDIENING

SENSOREN

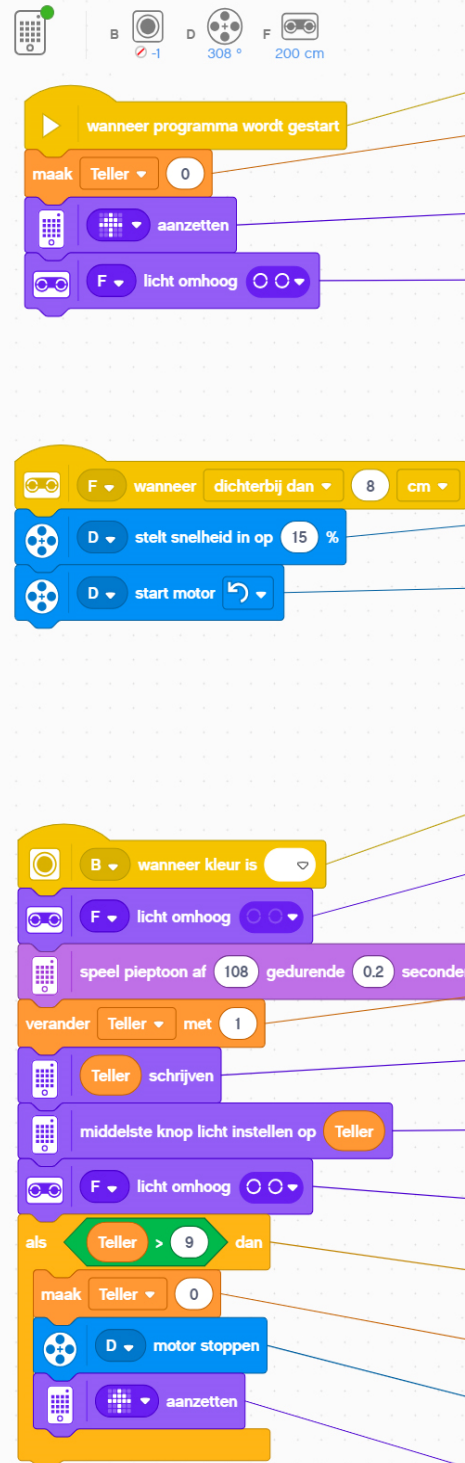
BEDIENERS

VARIABLEN

MIJN BLO...

SBC Uitbreiding

SBC Basis



Als het programma opstart, wordt deze code (stack) uitgevoerd.

Optioneel: Tellerwaarde op 0 zetten. (Dit hoeft eigenlijk niet, maar kan een goede gewoonte zijn voor later in andere programmeertalen.)

Pijl op scherm tonen die wijst naar de Afstandssensor.

"Ogen" van de Afstandssensor laten oplichten.

Telkens de Afstandssensor iets waarneemt dichterbij dan 8 cm, dan wordt deze code (stack) uitgevoerd.

Snelheid van motor D instellen op 15%

Motor D starten (deze motor zal blijven draaien tot we verder in het programma deze motor terug stoppen).

Telkens de Kleursensor iets wits ziet, dan wordt deze code (stack) uitgevoerd.

"Ogen" van de Afstandssensor doven.

De SPIKE Hub gedurende 0,2 sec een hoge pieptoon laten afspelen.

Hier laten we de waarde van de variabele met 1 verhogen. (= 1 erbij tellen.)

De waarde van onze Teller variabele op het scherm van de Hub plaatsen.

Laat de middelste knop van de SPIKE Hub oplichten met de kleur die overeenstemt met de waarde van onze Teller variabele.

"Ogen" van de Afstandssensor terug laten oplichten.

Als de waarde van de Teller groter is dan 9, dan wordt de code in dit C-Blok uitgevoerd, anders niet.

De waarde van de variabele Teller terug op 0 plaatsen.

Motor D terug stoppen.

Pijl op scherm tonen die wijst naar de Afstandssensor.

Mogelijke Leerplandoelen:

Algemene doelen en vaardigheden computationeel denken (CT) als instrument bij STEM

Specifieke vaardigheden bij SPIKE Prime Small Ball Contraption

Algoritmes en procedure: Leerlingen zetten procedures efficiënt in om tot oplossingen te komen.

De leerlingen kunnen met basiskennis aangereikt bij de start van de activiteit, komen tot een technische oplossing voor het bouwen van een machine.

Automatisering: Leerlingen passen standaardprocedures toe die rechtstreeks naar een oplossing leiden.

De leerlingen gebruiken standaard programmeerblokken als hoeksteen voor een programma en gebruiken automatisch een aantal blokken als ruggengraat om te komen tot het oplossen van een probleem.

Uitbreiding: De leerlingen gaan op zoek om een programma eenvoudiger te schrijven.

Omgaan met gegevens: Leerlingen analyseren, verzamelen en visualiseren gegevens. (uitbreiding)

De leerlingen berekenen de snelheid van de cyclus van de bal.

De leerlingen maken een grafiek als toepassing op de machine:

-aantal keren,

-aantal cycli per minuut,

-vergelijken van de gegevens met de machine van een medeleerling,

-aantal gepasseerde ballen/voorwerpen.

Debugging: Leerlingen sporen fouten op in algoritmes en codes.

De leerlingen bouwen en testen bij het schrijven van een programma elk onderdeel afzonderlijk vóór ze naar een volgende programmeerstap gaan.

De leerlingen kunnen terugkeren op hun stappen om fouten op te sporen in een programma.

Simulatie en modelleren: Leerlingen bootsen de werkelijkheid na.

De leerlingen herkennen de toepassing van de machine in de wereld om hen heen. Ze kunnen de transfer maken naar bijvoorbeeld een bowlingbaan (automatisch terugkeren van de ballen), een teller bij een toegangspoortje (attractie in pretpark, toegang toiletten), lotto (de 'getrokken' bal die passeren bij het lottospel), ...

Uitbreiding: de machine registreert het aantal rode, groene, blauwe, witte en gele ballen afzonderlijk. (Hiervoor zullen wel gekleurde ballen moeten gezocht worden, of kan met dit simuleren door de gekleurde LEGO stenen te gebruiken die wel in de SPIK Prime set aanwezig zijn.)