

Programmer 1: Variabelen en rekenen

Kaart 1a - Opdracht 1: Secondeteller

Opdracht

Bouw nu met deze blokken een teller. Tussen het tonen van ieder getal moet één seconde worden gewacht.

Oplossing



```
Scratch code for a counter starting at 0:  
- bij opstarten  
  - stel teller in op 0  
- de hele tijd  
  - toon nummer teller  
  - verander teller met 1  
  - pauzeer (ms) 1000
```

https://makecode.microbit.org/_Ji5bzg5bgA8d

Ik snap het

Ik kan dit programma zo aanpassen dat het begint te tellen bij 2 en daarna alleen even getallen laat zien (0, 2, 4 etc.)

Oplossing



```
Scratch code for a counter starting at 2:  
- bij opstarten  
  - stel teller in op 2  
- de hele tijd  
  - toon nummer teller  
  - verander teller met 2  
  - pauzeer (ms) 1000
```

https://makecode.microbit.org/_X4AFsT9xKH2A

Ik snap het

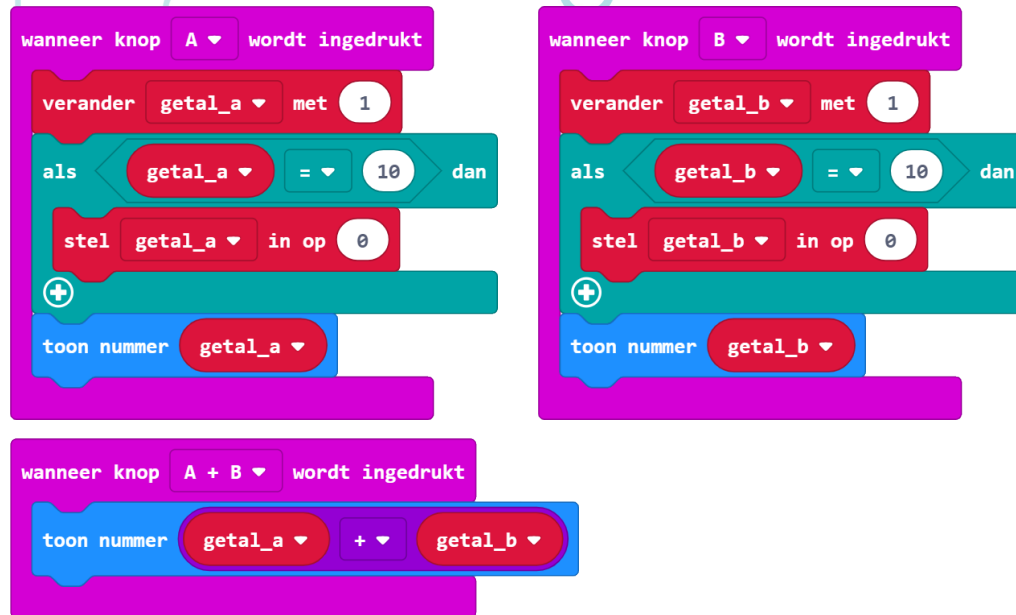
Ik kan vertellen wat er met de naam van de variabelen in het project gebeurt.

Antwoord

Als je de naam van een variabele verandert dan wordt deze verandering in het hele project doorgevoerd.

Kaart 1b - Opdracht 2: Rekenmachine

Het programma



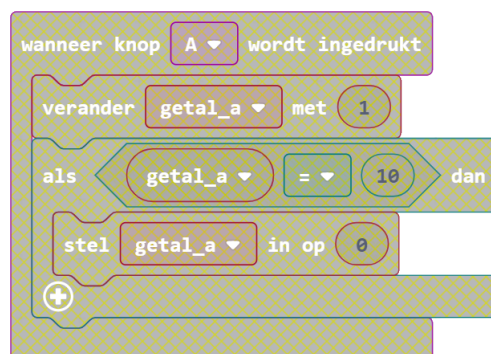
https://makecode.microbit.org/_XHkgqEX2pD5o

Achtergrond: Namen van variabelen

- Als in een programma veel verschillende variabelen worden gebruikt dan is het belangrijk dat leerlingen leren om variabelen een logische naam te geven.
- Voorkom spaties in de namen van variabelen. De JavaScript editor accepteert deze spaties niet.
- De naam van een variabele mag niet met een cijfer beginnen.
- Programmeurs gebruiken bij de namen van variabelen hoofdletters en kleine letters *getal_a*, zal door veel programmeurs als *getalA* worden geschreven.

Ik snap het 1

Toen je de code van knop A dupliceerde was de kopie vaag. Waarom?



Antwoord

Er was al een blok *on button A pressed*. Stel dat je twee van deze blokken zou hebben dan weet de micro:bit niet welke code uitgevoerd moet worden als op knop A wordt gedrukt.



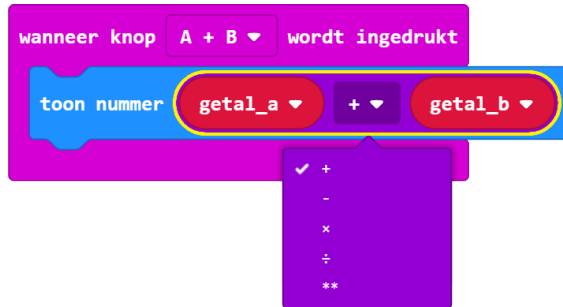
Het blok *de hele tijd* kan je wel vaker in één programma gebruiken.

Ik snap het 2

- In plaats van optellen kan ik de getallen ook vermenigvuldigen, aftrekken en delen.

Oplossing

Dit doe je door de operator in het rekenblokje aan te passen



Operator	Teken	Voorbeeld
Optellen	+	$6 + 2 = 8$
Aftrekken	-	$6 - 2 = 4$
Vermenigvuldigen	x	$6 \times 2 = 12$
Delen	÷	$6 / 3 = 2$
Machtsverheffen	**	$6 ** 2 = 6^2 = 36$

Over machtsverheffen

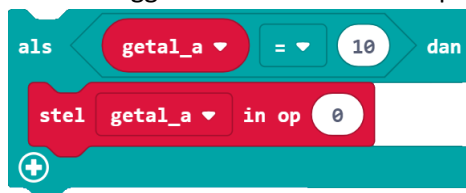
Bij machtsverheffen wordt een getal een opgegeven aantal keer met zichzelf vermenigvuldigd. Er zijn verschillende manieren om dit op te schrijven:

3 tot de macht 2	$3^2 = 9$	$3 \wedge 2 = 9$	$3 \times 3 = 9$	$3 ** 2 = 9$
3 tot de macht 3	$3^3 = 27$	$3 \wedge 3 = 27$	$3 \times 3 \times 3 = 27$	$3 ** 3 = 27$

Operator: [https://nl.wikipedia.org/wiki/Operator_\(wiskunde\)](https://nl.wikipedia.org/wiki/Operator_(wiskunde))

Ik snap het 3

- Ik kan uitleggen wat dit deel van het programma doet



Antwoord

Dit deel zorgt ervoor dat het hoogste getal dat kan worden ingevoerd 9 is. In dit voorbeeld hebben we hiervoor gekozen om te voorkomen dat de tekst tijdens het invoeren gaat scrollen.

De voorwaarde kan je ook op twee andere manieren opgeven:



Programmer 2: Debuggen en algoritmes

Kaart 2a - Opdracht 1: Foutje

Ik kan uitleggen ...

... of dit een fout in het algoritme of een syntaxfout is.

Antwoord



```
bij opstarten
  stel mijnGetal in op "2"
  toon nummer mijnGetal
```

https://makecode.microbit.org/_gCHLTC4MWEmm

De variabele *mijnGetal* wordt eerst gevuld met een tekst (string), waarna wordt geprobeerd om deze tekst te laten zien met het blokje *show number*. Dit blokje kan echter alleen overweg met getallen.

Voorbeeld van een fout in het algoritme

Bij dit programma wordt de tekst "Hallo" niet getoond omdat de blokjes *show string* en *set mijnTekst to* in de verkeerde volgorde staan.



```
bij opstarten
  toon tekens mijnTekst
  stel mijnTekst in op "Hallo"
```

https://makecode.microbit.org/_gpb2P1LpihFu

Over het algemeen kan je het volgende zeggen:

- Bij een *syntaxfout* doet het programma het niet.
- Bij een *fout in het algoritme* doet het programma niet wat de programmeur wilde.

Ik snap het

... want ik kan het programma op twee manieren aanpassen zodat er wel een 2 op het display komt.

Oplossing 1

Gebruik een het blok *toon tekens* om de inhoud van een variabele van het type String te laten zien



```
bij opstarten
  stel mijnGetal in op "2"
  toon tekens mijnGetal
```

https://makecode.microbit.org/_Php8Wo2y1ev3

Oplossing 2

Vul de variabele met een waarde van het type Number (Getal)

```

bij opstarten
  stel mijnGetal in op 2
  toon nummer mijnGetal
  
```

<https://makecode.microbit.org/iwpWW2L2bU99>

Oplossing 3 (voor de specialisten)

Je kan van een *tekenreeks* een *getal* maken. Hieronder zie je hoe dat kan:

```

bij opstarten
  stel mijnGetal in op "2"
  stel mijnEchteGetal in op parse to number mijnGetal
  toon nummer mijnEchteGetal
  
```

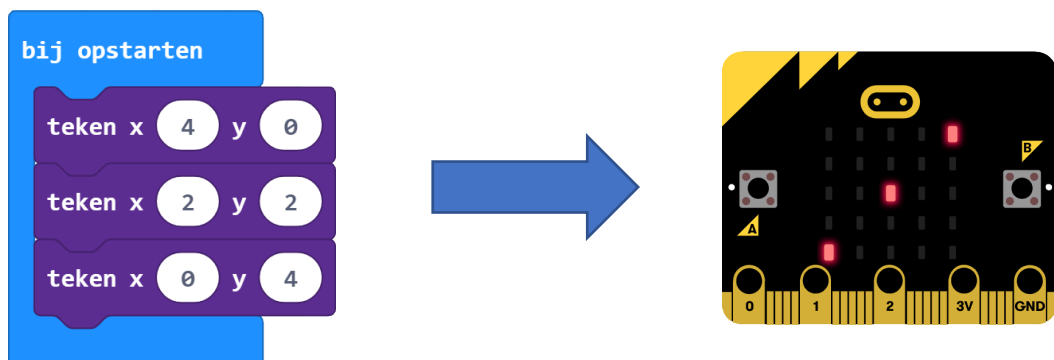
<https://makecode.microbit.org/7Df7ijHyXYo5>

Als een variabele eenmaal is gevuld met een string of met een getal dan kan je het type niet meer veranderen. Je kan wel een nieuwe variabele (*mijnEchteGetal*) maken en een parse (converteer) blok gebruiken om van de string een getal (zie hieronder) te maken.

Kaart 2b - Opdracht 2: Plot erop los

Tip blokje plot x y

Laat de leerlingen eventueel eerst het blokje *teken x y* ontdekken met dit programma.



Bouw het programma



```

bij opstarten
  stel teller in op 0

de hele tijd
  stel teller in op teller + 1
  teken x teller y 0
  pauzeer (ms) 100
  Wis scherm
  als teller = 5 dan
    stel teller in op 0

```

https://makecode.microbit.org/_OKY8gyCAYfyz

Ik kan vertellen ...

... wat het 'foute' programma wel doet en wat niet.

Antwoord

Het looplicht begint telkens bij het tweede ledje in plaats van bij het eerste.

Bij het zoeken naar fouten is het belangrijk dat je zo precies mogelijk beschrijft wat je ziet en wat er fout gaat.

Ik snap het

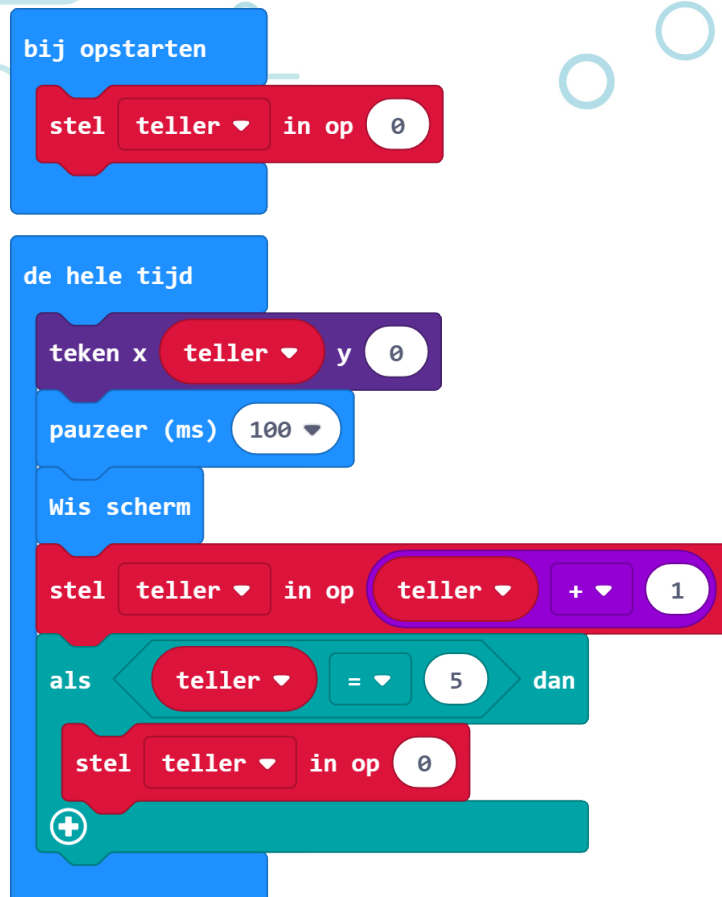
Ik heb het programma zo aangepast dat het wél doet wat de bedoeling is.

Wat gaat er fout?

Het gaat fout met het eerste blokje in de lus *de hele tijd*. Hier wordt de variabele *teller* al met één verhoogd terwijl het ledje op (0,0) nog moet worden aangezet. Hierna staan twee mogelijke oplossingen.

Oplossing 1

Heel veel programma's werken met dit soort tellers en het is gebruikelijk om de teller pas op het einde van een lus op te hogen. Dit is ook de 1^e oplossing



```

bij opstarten
  stel teller in op 0

de hele tijd
  teken x teller y 0
  pauzeer (ms) 100
  Wis scherm
  stel teller in op teller + 1
  als teller = 5 dan
    stel teller in op 0

```

https://makecode.microbit.org/_hueP9ECDREhW

Oplossing 2

Hierbij blijft de volgorde van de blokken (syntax) hetzelfde, maar beginnen we niet te tellen bij 0, maar bij -1.



```

bij opstarten
  stel teller in op -1

```

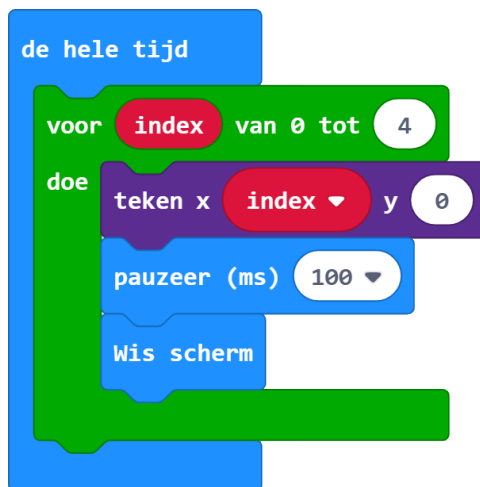


```
de hele tijd
  stel teller in op teller + 1
  teken x teller y 0
  pauzeer (ms) 100
  Wis scherm
  als teller = 4 dan
    stel teller in op -1
```

<https://makecode.microbit.org/XoqKR4c38U8b>

Een ander algoritme

In plaats van het 'zelf' ophogen van de teller en telkens met een *als ... dan* blok controleren of deze de maximale waarde heeft bereikt kan je ook gebruik maken van het *voor ... van 0 tot ...* blok. (deze zogenaamde 'lus' wordt in kaart 3 uitgelegd)



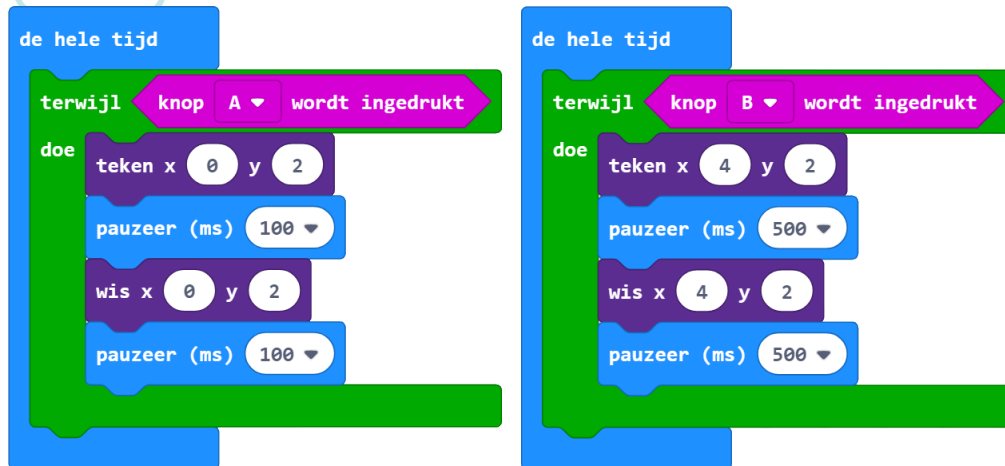
```
de hele tijd
  voor index van 0 tot 4
  doe
    teken x index y 0
    pauzeer (ms) 100
  Wis scherm
```

<https://makecode.microbit.org/Aff4veiqJcmy>

Programmer 3a: Lussen

Kaart 3a - Opdracht 1: Multitasken

Het programma



https://makecode.microbit.org/_3Dod9Ubj9aqU

Met *multitasken* bedoelen we dat een computer of een mens verschillende taken tegelijk uitvoert. Voor wat betreft de computer moet worden opgemerkt dat het lijkt alsof dit zo is. In de praktijk worden de verschillende taken in kleine stukjes gesplitst die razendsnel achter elkaar worden uitgevoerd.



Doordat het blokje *de hele tijd* vaker gebruikt kan worden, is het eenvoudig om de micro:bit verschillende zaken tegelijk te laten doen. Alle andere 'hoofdblokken' mag je maar één keer in een programma gebruiken. In de editor zijn deze fouten gemarkeerd met een geel

raster.



https://makecode.microbit.org/_39cLcdadCHx8

Ik snap waarom...

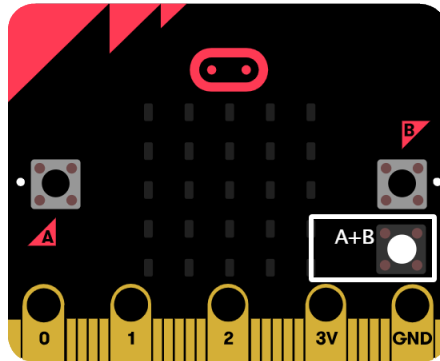
... dit programma niet (helemaal) werkt in de simulator.

Antwoord

In de simulator is (nog) geen mogelijkheid om de twee knoppen tegelijk in te drukken (de muis zou dan op twee plaatsen tegelijk moeten zijn).

wanneer knop **A + B** wordt ingedrukt

Voor het blok *wanneer knop A + B wordt ingedrukt* verschijnt in de simulator een knop waarmee het indrukken van de twee toetsen tegelijk wordt gesimuleerd:

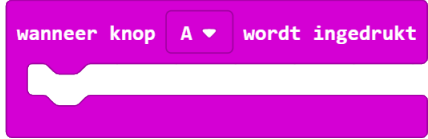



Dit werkt echter niet bij de blokjes *knop A wordt ingedrukt* en *knop B wordt ingedrukt*.

Ik snap het verschil ...

... tussen deze blokken:

Antwoord

	
<p>Reageert als de knop wordt ingedrukt en daarna wordt losgelaten.</p>	<p>Reageert direct als dit blokje 'aan de beurt is' in het programma en de knop is ingedrukt.</p>

Kaart 3b - Opdracht 2: Knippen (maar niet te vaak)

Het programma



https://makecode.microbit.org/_aLC3ribiqEjc

Ik snap het

Wordt met het *tot* in dit blok *tot* of *tot en met* bedoeld?


Antwoord

In het blokje doet de vier ook mee. Samen met de nul wordt de lus dus vijf keer doorlopen. Taalkundig zou het dus *tot en met* moeten zijn.



Kaart 3c - Opdracht 3: Playlist

Het programma



https://makecode.microbit.org/_cTkdGW4fee25

Dit is een moeilijke kaart en we hebben hem opgenomen om alle soorten lussen te kunnen behandelen.

Het is ook voor het eerst dat we een uitstapje maken naar de JavaScript editor.

Ik snap het

Ik kan het programma zo aanpassen dat de lijst met tonen 4 keer wordt afgespeeld als ik op *knop A* druk.

Oplossing



https://makecode.microbit.org/_e5pKTxKbhEg8

Voor de specialisten

```

bij opstarten
  stel mijnLijst in op
    array of
      250
      500
      750
      1000
      1250
      1500
  
```

```

wanneer knop A wordt ingedrukt
  voor element mijnInhoud van mijnLijst
  voer uit
    speel toon mijnInhoud voor 1 beat
  
```

https://makecode.microbit.org/_dEaX6z7g0dMH

Let op Bij het werken met de JavaScript (tekst) versie van het programma mag geen tyfout gemaakt worden, zoals bijvoorbeeld `]` vervagen door `,` , of een punt gebruiken i.p.v. een komma. Dit leidt in de meeste gevallen tot syntaxfouten waardoor het programma niet meer (goed) kan worden uitgevoerd, of niet meer naar blokken kan worden teruggezet.

Dit kan ook

In dit voorbeeld zijn de tonen in de reeks steeds een veelvoud van 250. Hierom kan in dit voorbeeld hetzelfde resultaat worden bereikt met dit programma:

```

wanneer knop A wordt ingedrukt
  voor index van 0 tot 5
  doe
    speel toon 250 + index * 250 voor 1 beat
  
```

https://makecode.microbit.org/_eMWhjf3FV8xY

Programmer 4: Voorwaarden

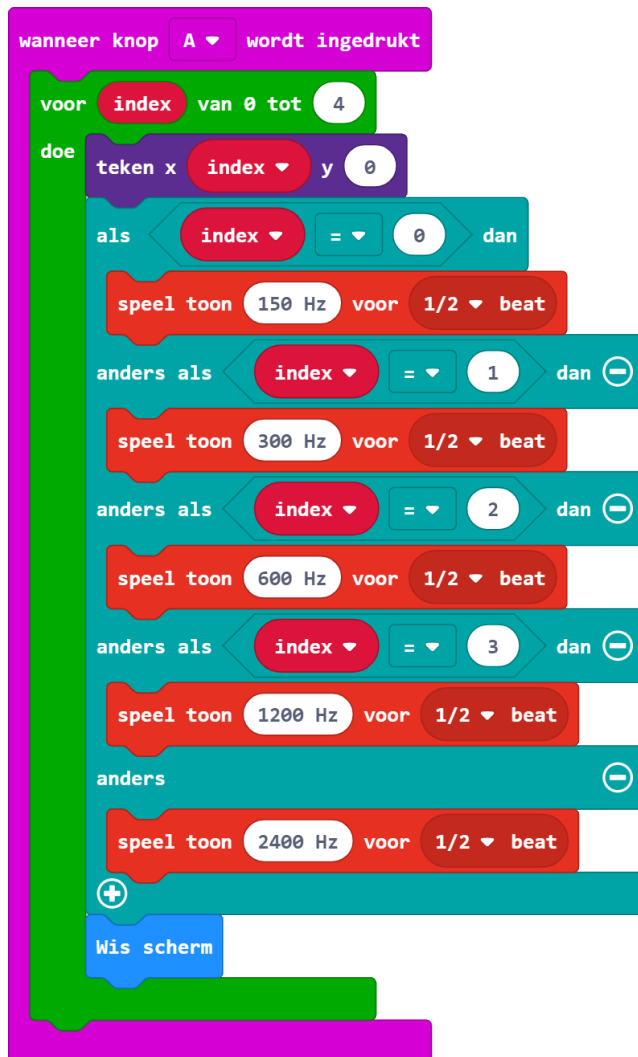
Tip Computational Thinking

In de leerlijn Programmeren in het VO staan leuke unplugged¹ oefeningen om kennis te maken met *als – dan – anders als – anders*.

- Middenbouw: [https://maken.wikiwijs.nl/74282/Programmeren in het PO#!page-1838209](https://maken.wikiwijs.nl/74282/Programmeren%20in%20het%20PO#!page-1838209)
- Bovenbouw: [https://maken.wikiwijs.nl/74282/Programmeren in het PO#!page-1838211](https://maken.wikiwijs.nl/74282/Programmeren%20in%20het%20PO#!page-1838211)

Kaart 4a - Opdracht 1: Toonladder

Het programma



```

wanneer knop A wordt ingedrukt
  voor index van 0 tot 4
    doe
      teken x index y 0
      als index = 0 dan
        speel toon 150 Hz voor 1/2 beat
      anders als index = 1 dan
        speel toon 300 Hz voor 1/2 beat
      anders als index = 2 dan
        speel toon 600 Hz voor 1/2 beat
      anders als index = 3 dan
        speel toon 1200 Hz voor 1/2 beat
      anders
        speel toon 2400 Hz voor 1/2 beat
    +
  Wis scherm
  
```

https://makecode.microbit.org/_amFTefgxR3dm

Het blokje voor ... van 0 tot ...

Met dit programmaatje kan je de werking van het blokje laten zien

¹ Unplugged activiteiten zijn oefeningen over Computational Thinking waarbij geen computer nodig is

```

de hele tijd
  voor index van 0 tot 4
  doe
    toon nummer index
    pauzeer (ms) 100
  
```

https://makecode.microbit.org/_8FYfDET1wFew

Ik snap het

- De variabele *teller* telt van 0 t/m 4. Waarom staat dit dan niet in het programma?

```

index = 4

```

Antwoord

Dit is niet nodig omdat dit bij de voorwaarde *anders* is afgevangen.

Door een extra *anders als* toe te voegen en *anders* te verwijderen zou je dit wel nodig hebben:

```

wanneer knop A wordt ingedrukt
  voor index van 0 tot 4
  doe
    teken x index y 0
    als index = 0 dan
      speel toon 150 Hz voor 1/2 beat
    anders als index = 1 dan
      speel toon 300 Hz voor 1/2 beat
    anders als index = 2 dan
      speel toon 600 Hz voor 1/2 beat
    anders als index = 3 dan
      speel toon 1200 Hz voor 1/2 beat
    anders als index = 4 dan
      speel toon 2400 Hz voor 1/2 beat
    Wis scherm
  
```

https://makecode.microbit.org/_er5hY8eEyThe

Ander algoritme 1

Omdat er een rekenkundig verband is tussen de opeenvolgende tonen zijn er ook andere manieren (algoritmes) om hetzelfde resultaat te bereiken. Eén hiervan is deze:

```

wanneer knop A wordt ingedrukt
  stel mijnToon in op 150
  voor index van 0 tot 4
  doe
    teken x index y 0
    speel toon mijnToon voor 1/2 beat
  wis scherm
  stel mijnToon in op 2 x mijnToon
  
```

https://makecode.microbit.org/_9qCJ6r5cmDXf

Kaart 4b - Opdracht 2: EN en/of spelletje

Het programma

```

wanneer knop A wordt ingedrukt
  wis x x y y
  verander x met 1
  als x = 5 dan
    stel x in op 0
  teken x x y y

wanneer knop B wordt ingedrukt
  wis x x y y
  verander y met 1
  als y = 5 dan
    stel y in op 0
  teken x x y y

de hele tijd
  als x = 2 en y = 2 of y = 0 of y = 4 dan
    wissel x x y y
    pauzeer (ms) 100
  
```

https://makecode.microbit.org/_gK4ghvVHa84g

```

de hele tijd
als x = 2 en y = 2 dan
  wissel x x y y
  pauzeer (ms) 100
anders als y = 0 of y = 4 dan
  wissel x x y y
  pauzeer (ms) 100
  
```

https://makecode.microbit.org/_WJdFWUDt9EUo

Nog een manier

```

de hele tijd
als x = 2 dan
  als y = 2 dan
    wissel x x y y
    pauzeer (ms) 100
  +
  +
  als y = 0 dan
    wissel x x y y
    pauzeer (ms) 100
  +
  als y = 4 dan
    wissel x x y y
    pauzeer (ms) 100
  +
  
```

https://makecode.microbit.org/_RDAem4czKbvp

Ik snap het

De zin *Kim of Jan en Fatima* hebben taart kan je op twee manieren uitleggen. Ik kan het verschil uitleggen tussen:



Oude blokjes?

Inderdaad, hierboven en hieronder zie je nog de blokjes zoals die in de oude editor werden gebruikt. Deze had de mogelijkheid om lange samengestelde voorwaarden op een overzichtelijke manier weer te geven, onder elkaar in plaats van alles naast elkaar. In het Nederlands heette dit *Externe invoeren*. In het Engels was het *External Inputs*.

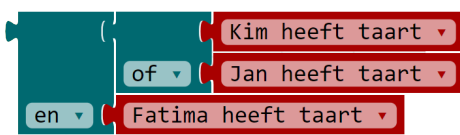
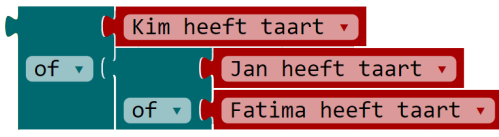
Als je ook vindt dat deze mogelijkheid weer terug moet komen, ga dan naar de Github-site van de ontwikkelaars en maak je wensen kenbaar:

<https://github.com/Microsoft/pxt-microbit/issues/1698>

We hebben de oude blokjes hier gebruikt omdat ze zo lekker duidelijk waren

Antwoord

Dit is een typisch voorbeeld waarbij de programmeur zich moet afvragen wat precies de gewenste logica is. Hieronder zie je hoe de zin *Kim of Jan en Fatima hebben taart* duidelijker kan worden geschreven.

Oude blokjes										
	<table border="1"> <tr> <td rowspan="2">EN</td> <td>OF</td> <td>Kim heeft taart</td> </tr> <tr> <td></td> <td>Jan heeft taart</td> </tr> <tr> <td></td> <td></td> <td>Fatima heeft taart</td> </tr> </table>	EN	OF	Kim heeft taart		Jan heeft taart			Fatima heeft taart	<p>Met een komma:</p> <ul style="list-style-type: none"> Kim of Jan, en Fatima hebben taart <p>Met haakjes:</p> <ul style="list-style-type: none"> (Kim of Jan) en Fatima hebben taart
EN	OF		Kim heeft taart							
		Jan heeft taart								
		Fatima heeft taart								
	<table border="1"> <tr> <td rowspan="2">OF</td> <td></td> <td>Kim heeft taart</td> </tr> <tr> <td>EN</td> <td>Jan heeft taart</td> </tr> <tr> <td></td> <td></td> <td>Fatima heeft taart</td> </tr> </table>	OF		Kim heeft taart	EN	Jan heeft taart			Fatima heeft taart	<p>Met een komma:</p> <ul style="list-style-type: none"> Kim of, Jan en Fatima hebben taart <p>Met haakjes:</p> <ul style="list-style-type: none"> Kim of (Jan en Fatima) hebben taart
OF			Kim heeft taart							
	EN	Jan heeft taart								
		Fatima heeft taart								

Voor de specialisten

Dit probleem komt ook voor bij rekenen. Bij een som als $3 + 2 \times 4$ is echter vastgelegd in welke volgorde de berekening moet worden uitgevoerd (oude ezelsbruggetje Mijnheer Van Dale Wacht op Antwoord).

Bij logica geldt de regel NAO:

1. Not
2. And
3. Or

Dit speelt een rol bij tekstuele programmeertalen waar de regel zo geschreven wordt:

```
(Kim_heeft_taart || Jan_heeft_taart && Fatima_heeft_taart)
```

Operator	Betekenis
	OR
&&	AND

Kaart 4c - Opdracht 1: Wie drukte als eerste?

Het programma



https://makecode.microbit.org/_9iD909C8YAkx

Vraag

- Wat gebeurt als je knop A ingedrukt houdt en daarna op knop B drukt?
- Wat gebeurt als je knop B ingedrukt houdt en daarna op knop A drukt?

Antwoord

In het eerste geval blijft de *A* staan, het *anders dan* gedeelte van het blokje wordt niet meer uitgevoerd (wat ook de bedoeling is).

In het tweede geval verandert de *B* in een *A* (niet de bedoeling).

Oplossing

```

de hele tijd
als knop A wordt ingedrukt dan
als niet knop B wordt ingedrukt dan
  toon tekens "A"
anders als knop B wordt ingedrukt dan
  als niet knop A wordt ingedrukt dan
    toon tekens "B"
  
```

https://makecode.microbit.org/_diY6665Md3Tt

Snap ik het?

Ik kan voorspellen wat er op het display komt te staan als op de knop wordt gedrukt.

```

de hele tijd
als niet niet knop A wordt ingedrukt dan
  toon nummer 1
anders
  toon nummer 0
  
```

Antwoord

Als op *knop A* wordt gedrukt komt er een *1* te staan.

Het blokje *knop A wordt ingedrukt* geeft *Waar (True)* als er op de knop wordt gedrukt.

Daarvoor staat twee keer een *niet* blokje die de waarde van *knop A wordt ingedrukt* telkens 'omdraaien'. Dit heet officieel *inverteren*.

Het werkt net als met een muntje: als je het twee keer omdraait dan ligt het weer hetzelfde. Drie keer omdraaien geeft hetzelfde effect als één keer omdraaien. De twee *niet* blokjes doen dus uiteindelijk niets.

Programmer 5: Meet het lichtniveau

Kaart 5a - Opdracht 1: Plot staafdiagram

Het programma

```
de hele tijd
  voor index van 0 tot 24
  doe
    plot bar graph of index
    up to 24
    pauzeer (ms) 200
```

https://makecode.microbit.org/_Y55Xi8aaq99d

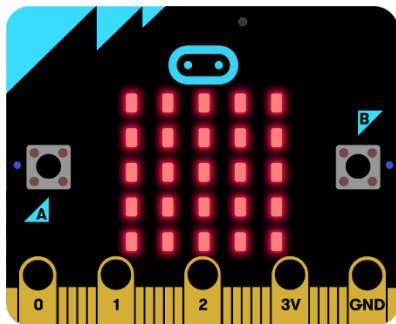
Ik kan voorspellen ...

... hoe het staafdiagram er uitziet als ik dit programmeer.

```
plot bar graph of 100
up to 25
```

Antwoord

De maximale waarde die wordt verwacht is 25. Bij 100 gaan dus alle leds branden:



Kaart 5a - Opdracht 2: Meet het lichtniveau

Het programma

```
de hele tijd
  plot bar graph of lichtniveau
  up to 255
```

https://makecode.microbit.org/_LdiYHRhUs0yP

Hoe kan het dat het display tegelijk licht kan geven licht kan meten?

In het totaal worden er negen leds gebruikt om licht te meten. Dit doen ze in 4 milliseconde (vier duizendste van een seconde). Van iedere seconde geven deze leds dus 99,6% licht en zijn ze 0,4% van de tijd bezig met het meten van licht. Dit gaat zo snel dat je niet ziet dat de leds tijdens de lichtmeting even uit zijn.

De lichtsensor is overigens het gevoeligst voor rood licht omdat de leds het gevoeligst zijn voor de kleur licht die ze kunnen geven.

1.1	1.2	1.3
2.2	2.3	2.1
3.3	3.1	3.2

Programmer 6: Versnelling en beweging

Kaart 6a - Opdracht 1: X-moves

Het programma



```
de hele tijd
  plot staafdiagram van versnelling (mg) x
  tot en met 1023
```

https://makecode.microbit.org/_MJ5MUw3pLLH8

Ik weet waarom ...

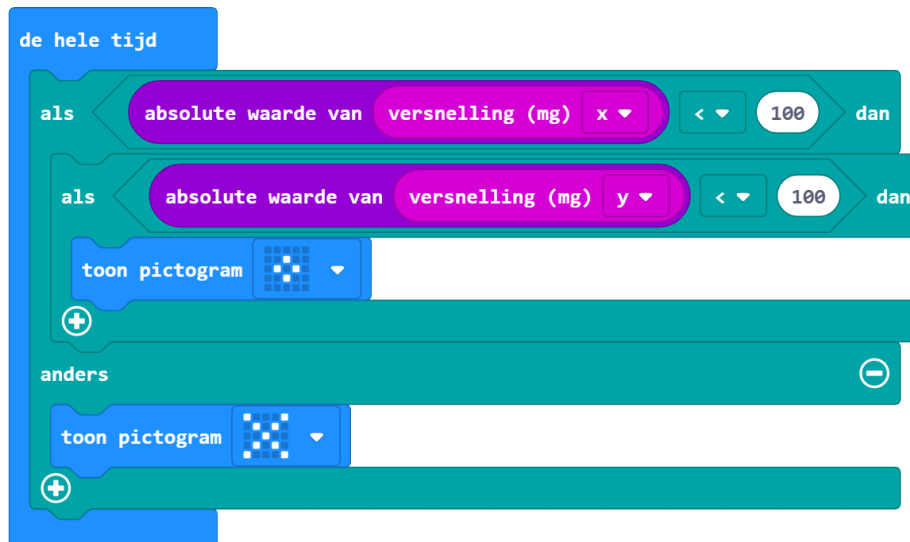
... in dit programma bij *up to* de waarde 1023 is ingevuld.

Antwoord

Dit is omdat 1023 de maximale waarde is die het blokje *versnelling (mg)* kan teruggeven.

Kaart 6b - Opdracht 2: Waterpas

Het programma



```
de hele tijd
  als absolute waarde van versnelling (mg) x < 100 dan
    als absolute waarde van versnelling (mg) y < 100 dan
      toon pictogram
    anders
      toon pictogram
```

<https://makecode.microbit.org/fPsVq7332Wki>

Ik weet waarom ...

... voor de waterpasstand wordt gecontroleerd of de absolute waarde van de X-versnelling en die van de Y-versnelling kleiner zijn dan 100.

Antwoord

Dit is omdat de nauwkeurigheid anders te groot zou zijn. De waarde van de versnelling kan namelijk variëren van -1023 tot +1023.

Op deze manier is de werking van het waterpas duidelijk. Er is natuurlijk niets op tegen om de nauwkeurigheid groter te maken, bijvoorbeeld 50.

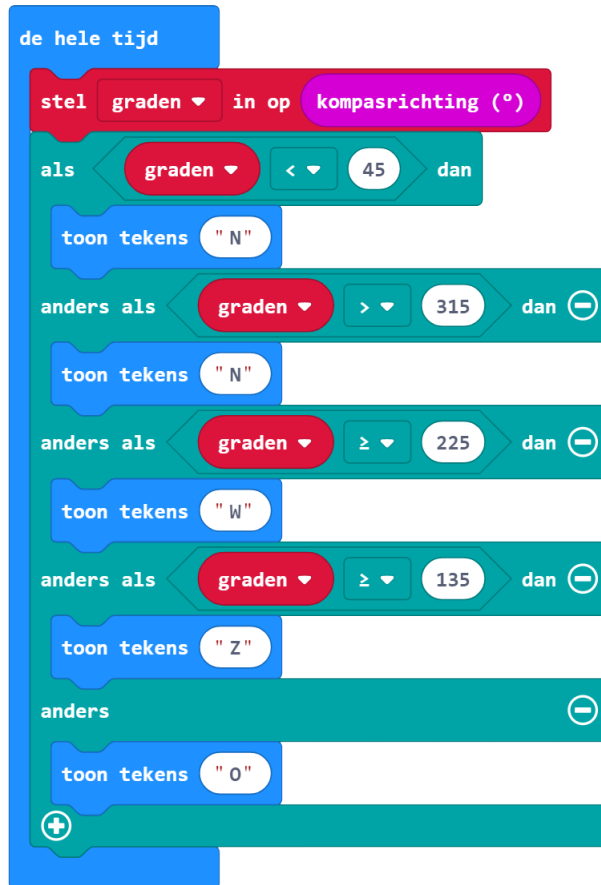
Tip: Maak er een spel van

Je kan de micro:bit waterpas gebruiken om er een spel mee te spelen waarbij één of meer kinderen de micro:bit moeten verplaatsen en daarbij horizontaal moeten houden. Het

wordt dan interessant om het programma zo aan te passen dat de leds gaan knipperen als de micro:bit te schuin wordt gehouden.

Kaart 6c - Opdracht 3: Maak een kompas

Het programma



```
de hele tijd
  stel graden in op kompasrichting (°)
  als graden < 45 dan
    toon tekens "N"
  anders als graden > 315 dan
    toon tekens "N"
  anders als graden ≥ 225 dan
    toon tekens "W"
  anders als graden ≥ 135 dan
    toon tekens "Z"
  anders
    toon tekens "0"
```

<https://makecode.microbit.org/EeP2xoVajbUF>

Ik snap het

Als we de windrichtingen opnoemen dan doen we dit meestal in de volgorde Noord, Oost, Zuid, West (met de klok mee).

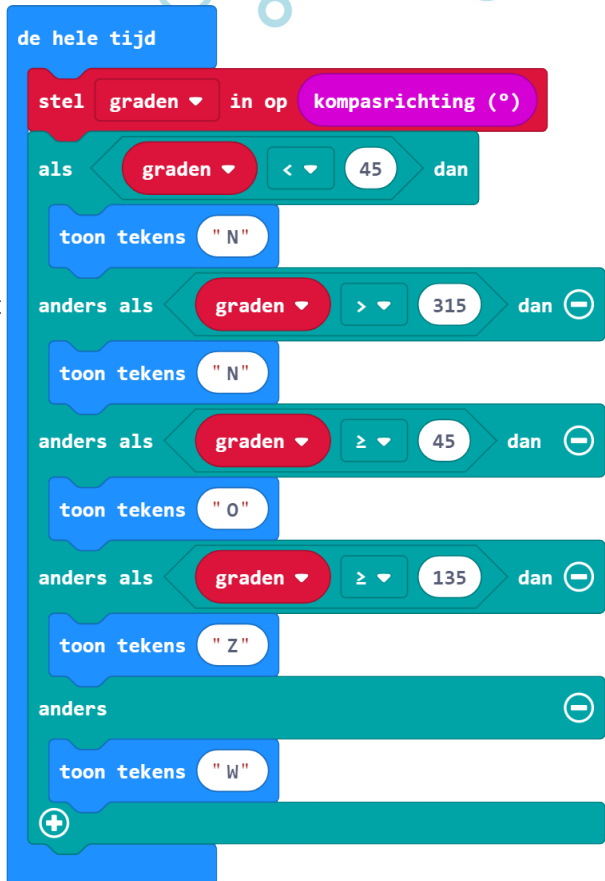
Waarom staan ze in het programma in de omgekeerde volgorde (Noord, West, Zuid, Noord)?

Antwoord

Stel dat het programma anders hadden opgebouwd.

In dit voorbeeld is de voorwaarde voor *Noord* nog steeds juist.

De voorwaarde voor *Oost* lijkt juist, maar stel dat het kompas richting *Zuid* (180°) wijst dan gaat het fout. 180 is namelijk ook groter dan 45 en op het display komt *Oost* te staan.



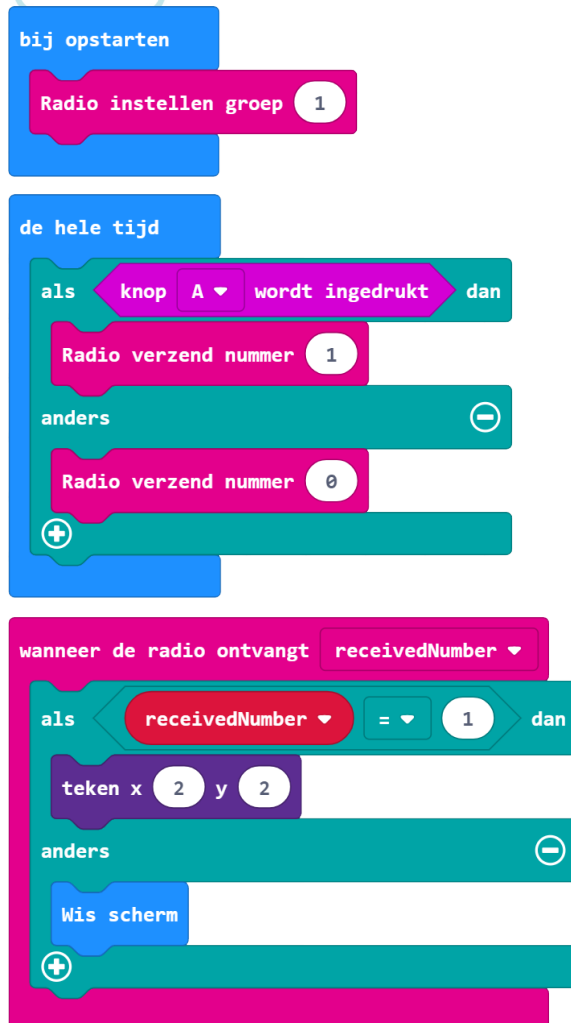
```
de hele tijd
  stel graden in op kompasrichting (°)
  als graden < 45 dan
    toon tekens "N"
  anders als graden > 315 dan
    toon tekens "N"
  anders als graden ≥ 45 dan
    toon tekens "O"
  anders als graden ≥ 135 dan
    toon tekens "Z"
  anders
    toon tekens "W"
```

https://makecode.microbit.org/_42xPPpC8R3xW

Programmer 7: Radio

Kaart 7a - Opdracht 1: Radioled

Het programma



```

bij opstarten
  Radio instellen groep 1

de hele tijd
  als knop A wordt ingedrukt dan
    Radio verzend nummer 1
  anders
    Radio verzend nummer 0

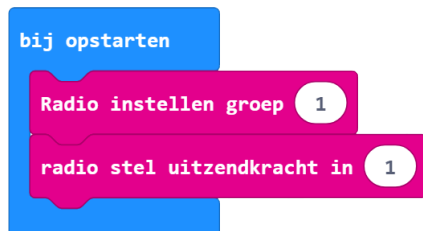
wanneer de radio ontvangt receivedNumber
  als receivedNumber = 1 dan
    teken x 2 y 2
  anders
    Wis scherm

```

https://makecode.microbit.org/_M8i38q1va7vd

Kaart 7b - Opdracht 2: Speurneus

Het programma van de zender



```

bij opstarten
  Radio instellen groep 1
  radio stel uitzendkracht in 1

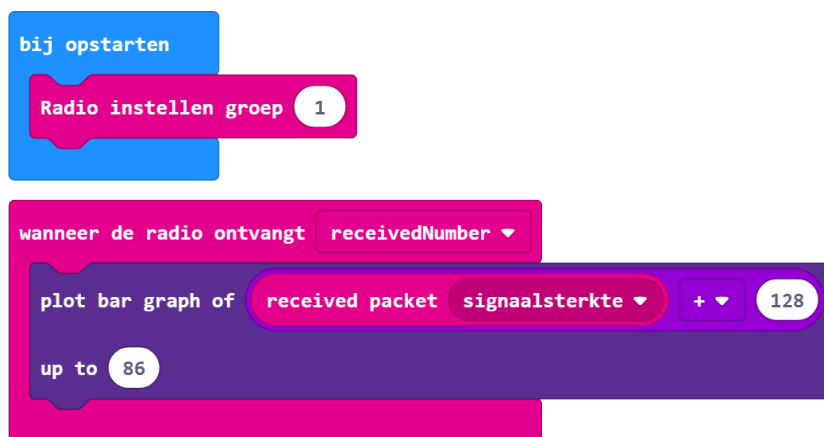
```



```
de hele tijd
  voor index van 0 tot 9
    doe
      Radio verzend nummer index
      toon nummer index
  pauzeer (ms) 100
```

<https://makecode.microbit.org/i2aPa8ficUf1>

Het programma van de ontvanger



```
bij opstarten
  Radio instellen groep 1

wanneer de radio ontvangt receivedNumber
  plot bar graph of received packet signaalsterkte + 128
  up to 86
```

<https://makecode.microbit.org/H1zdqwLVK7zm>

Vraag

Waarom moet de micro:bit eerst in een envelop?

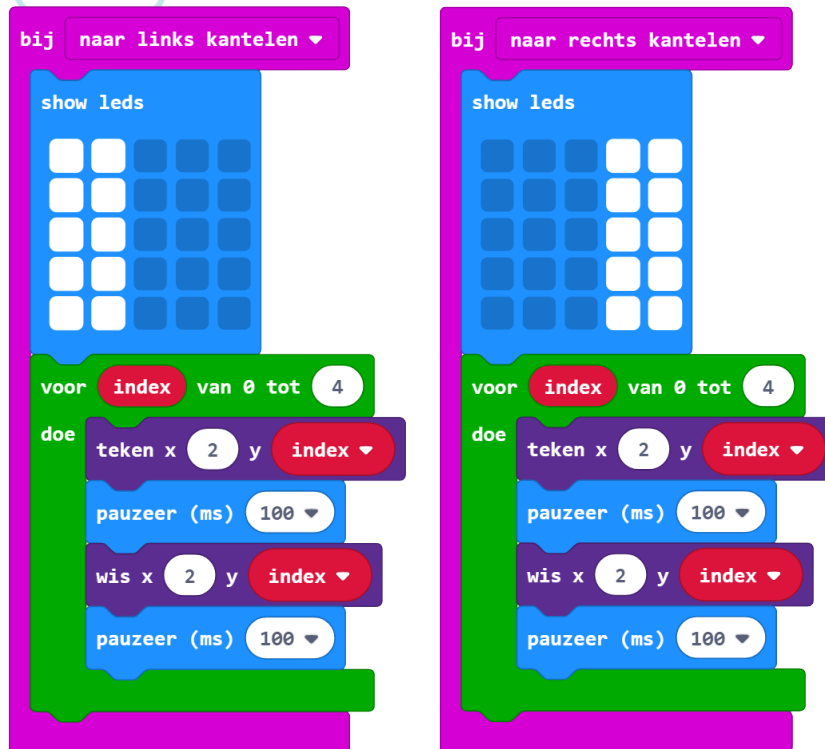
Antwoord

Hiermee voorkom je de kans op kortsluiting als je de micro:bit direct in aluminiumfolie zou wikkelen.

Programmer 8: Functies

Kaart 8a - Opdracht 1: Van links naar rechts

Het programma



```

bij naar links kantelen
show leds
voor index van 0 tot 4
doe
  teken x 2 y index
  pauzeer (ms) 100
  wis x 2 y index
  pauzeer (ms) 100

bij naar rechts kantelen
show leds
voor index van 0 tot 4
doe
  teken x 2 y index
  pauzeer (ms) 100
  wis x 2 y index
  pauzeer (ms) 100

```

https://makecode.microbit.org/_P089fEMuxL5R

Met functie

The image shows three Micro:bit code blocks. The top two are event-driven blocks: 'bij naar links kantelen' and 'bij naar rechts kantelen'. Both call a function named 'mijnFunctie' and then 'show leds'. The 'show leds' block displays a 5x5 grid of LEDs. In the 'naar links' version, the first two columns are lit (white), and the rest are dark (blue). In the 'naar rechts' version, the last two columns are lit. The third block is a function definition for 'mijnFunctie'. It uses a 'voor' loop from index 0 to 4. Inside the loop, it sets the LED at x=2, y=index to 'teken' (white) and then 'wis' (dark) after a 100ms pause.

```
function mijnFunctie
  voor index van 0 tot 4
  doe
    teken x 2 y index
    pauzeer (ms) 100
    wis x 2 y index
    pauzeer (ms) 100
```

https://makecode.microbit.org/_chLMA7H83YvL

Programmer 9: Pins

Kaart 9a - Krijgt de plant genoeg water?

1^e versie van het programma

```

de hele tijd
  plot bar graph of lees analoog pin P0
  up to 1023
  
```

https://makecode.microbit.org/_MpW2cgYFv69P

2^e versie van het programma

```

de hele tijd
  schrijf analoog pin P1 naar 1023
  stel mijnWaarde in op lees analoog pin P0
  schrijf analoog pin P1 naar 0
  plot bar graph of mijnWaarde
  up to 1023
  pauzeer (ms) 5000

wanneer knop A wordt ingedrukt
  toon nummer mijnWaarde
  
```

https://makecode.microbit.org/_Fw61XbVWsbUE

Deze kaart is gebaseerd op één van de voorbeelden op de website van de editor:

<https://makecode.microbit.org/projects/soil-moisture/code>

Over analoog en digitaal

- Een *analoog* signaal kan alle waarden aannemen. Zo zit tussen 7 en 8 bijvoorbeeld 7,5, maar ook 7,55 en 7,5000067 etc.
- Een *digitale* waarde laat zich in een getal met een vast aantal digits (cijfers) beschrijven.
Met digitale techniek wordt over het algemeen bedoeld dat de signalen maar twee waarden (betekenissen) kunnen hebben:



Ja	Nee
----	-----

Waar	Onwaar
True	False
Eén	Nul
Aan	Uit

Zie ook: https://wikikids.nl/Analoog_en_digitaal

Om over na te denken

Zijn de cijfers die je leerlingen geeft analoog of digitaal? En wat is dan een *zes min* of een *zeven plus*?

Kaart 9b – Schuifmuziek

Het programma

```

bij opstarten
  stel geluidAan in op waar

wanneer knop A wordt ingedrukt
  stel geluidAan in op niet geluidAan

de hele tijd
  stel mijnWeerstand in op lees analoog pin P1
  als geluidAan = waar dan
  als mijnWeerstand > 400 dan
    speel toon (Hz) mijnWeerstand
  anders
    rust (ms) 1/16 beat
  
```

<https://makecode.microbit.org/JPYLWUXpYi56>

Over weerstanden

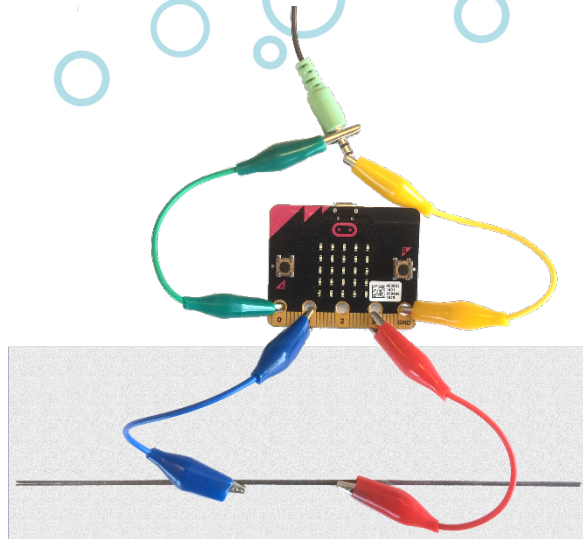
Weerstand is één van de drie elementen uit de Wet van Ohm. Deze wet beschrijft hoe elektrische spanning, stroom en weerstand zich tot elkaar verhouden:

$$Stroom = \frac{Spanning}{Weerstand}$$

De weerstand die bij deze opdracht wordt gebruikt bestaat uit het grafietpoeder (koolstof) uit een potlood.

Het is bij het 'tekenen' van de weerstand belangrijk dat er een 'vette' strook ontstaat.

De breedte van de streep moet ongeveer 2 mm zijn. Hoe breder je de streep maakt, hoe lager de weerstand en hoe kleiner het verschil in weerstand als je er met de klemmen over schuift.



Let op: Gebruik een 'echt' potlood. Er zijn ook potloden die geen grafiet bevatten.

Ik snap het

De basis van het programma waarmee de programmeur het idee heeft getest zag er zo uit:





<https://makecode.microbit.org/TW44Evi1AFMy>

Uiteindelijk zit er veel meer in het programma. Kan jij uitleggen wat het programma nog meer doet en welke blokjes hiervoor zorgen?

Antwoord

Ten opzichte van het basisprogramma heeft het uiteindelijke programma nog twee andere functionaliteiten:

- Aan/uit functie voor het geluid
- Onderdrukking van geluid als er geen contact wordt gemaakt (drempel)

	<p>Met de variabele <i>geluidAan</i> wordt geregeld of het geluid <i>Aan</i> of <i>Uit</i> staat.</p> <p>Bij de start van het programma wordt <i>geluidAan</i> gevuld met <i>True</i>.</p>
	<p>Met <i>knop A</i> wordt het geluid aan of uit gezet.</p>

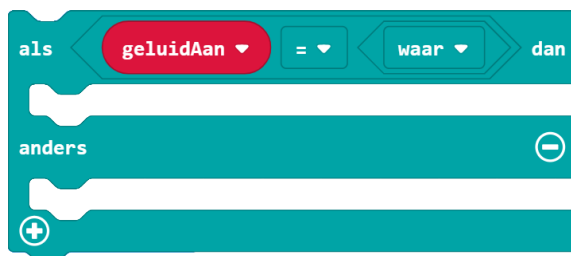


Er is alleen geluid als de variabele *geluidAan* de waarde *true* heeft EN de gemeten waarde van de weerstand (*mijnWeerstand*) groter is dan 400.

Het uitzetten van de toon wordt gerealiseerd met het blokje *speel toon*. Als *geluidAan* onwaar is, zorgt het blokje *rust (ms)* voor een stilte.

Verkorte versie als ... dan

In het eerste *als ... dan* blok hebben nu een blokje met een vergelijking gezet waarmee wordt gekeken of de variabele *geluidAan* de waarde *waar* heeft.



Omdat *geluidAan* een zogenaamde booleaanse variabele is en dus alleen maar de waarde waar of onwaar kan hebben, mag het blokje met de vergelijking worden weggelaten:



Aandachtspunten

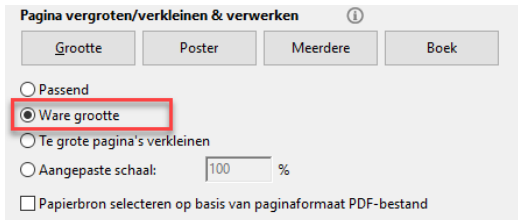
- Het kan zijn dat de drempel van 400 te hoog of te laag is. Experimenteer hiermee
- Als het programma niet werkt, controleer dan of de weerstand goed is aangesloten en ook werkt. Dit kan met het basisprogramma:



Bijlage 1: Kaarten afdrukken

De kaarten hebben een langwerpige vorm. Hierdoor kunnen ze ook op de (relatief) smalle Nederlandse schooltafeltjes naast de laptop of toetsenbord worden gelegd.

1. Druk de pdf in kleur af op wit papier. Kies er bij de printeropties voor dat het document op ware grootte wordt afgedrukt.



2. Snijd rondom 5 mm af
3. Vouw het papier dubbel
4. Lamineer de kaart. Gebruik als het even kan geen glanzend, maar mat lamineerplastic.

Als je meerdere kaarten moet afdrukken dan kan je ze dubbelzijdig afdrukken en na het afsnijden van de 5 mm doormidden snijden.