

# WEEKENDSCHOOL LESCYCLUS

## PROGRAMMEREN –

### BEGELEIDERSINSTRUCTIE LES 1B

#### 1. INTRODUCTIE

Dit document geeft een overzicht van les 1B van de lescyclus programmeren. Afhankelijk van de Weekendschoollocatie en het aantal lessen wordt dit lesonderdeel voorafgegaan door les 1A: het instrueren van een robot om een boterham met hagelslag te smeren.

#### 2. ONBEKEND MET SCRATCH?

Als je nog nooit met Scratch gewerkt hebt, dan moet je voor de les even Scratch leren. Als je al ervaring hebt met programmeren in een andere taal dan zul je in een half uurtje wel klaar zijn. Als programmeren nieuw is dan zal het zeker een paar uur kosten.

##### 2.1 HULP VOOR BEGELEIDERS

Als je ergens niet uitkomt, schroom dan niet om voor de les te bellen met Hans de Jong, 06 204 234 84. Kan in het algemeen elke dag tot 23 uur. Op de dag zelf kun je nog wel wat vragen, maar er is heel weinig tijd en daarom is het verstandig om alle vragen tevoren beantwoord te krijgen.

##### 2.2 SCRATCH LEREN

Begin hier: <https://scratch.mit.edu/> en kies *Probeer het* of *Bekijk voorbeeld*. Dan programmeer je in Scratch 2.0. In de les gebruiken we Scratch 1.4 en die kun je hier downloaden.

[https://scratch.mit.edu/scratch\\_1.4/](https://scratch.mit.edu/scratch_1.4/)

Introducties over Scratch vind je hier: <https://scratch.mit.edu/help/>

Je kunt niet precies dezelfde geluiden toevoegen etc., maar dat zou geen belemmering moeten zijn.

In de les gebruiken we een Raspberry Pi als computer. Die werkt bijna net zoals een Mac of Windows PC. Daar hoeft je je niet op voor te bereiden.

Mocht je zelf een Raspberry Pi 2B of 3B hebben dan kan ik je ook een image voor een micro SD kaart sturen. Als je wilt kun je ook een complete setup met Raspberry Pi, monitor, toetsenbord etc. lenen.

### 3. LES 1B

- We beginnen met een introductie over hoe ze moeten computeren zonder gezondheidsklachten te krijgen.
- Daar spelen we stukjes video af  
<http://www.hetklokhuis.nl/tv-uitzending/1142/Gezond%20computeren>  
Van 4:50 tot 8:35 en daarna van 11:07 tot 12:23.
- Dan laten we een filmpje zien over wat ze deze les gaan maken.  
<< link toevoegen als het filmpje is gemaakt >>
- We leggen we uit dat programmeren de één na laatste stap is. De laatste stap is testen. Maar het begint met het analyseren van het probleem en bedenken hoe je het aanpakt. Dat gaan we daarom eerst doen. Daarvoor gaan ze in groepjes met een begeleider zitten, schrijven op wat ze als grote stappen zien en bespreken dat.
- Daarna doen we de opgave  
*Een doolhofspel – red de prinses*
- Aan het eind van het eerste lesblok laten we ze rondlopen om bij elkaar te kijken en inspiratie op te doen voor het tweede lesblok, waarbij ze doorgaan met deze les 1B.

De les is een kleine variatie op de les van <http://www.codeuur.nl/lesmateriaal#smartlife>. Die zijn voor de leeftijd 8 – 12 jaar om in 60 – 90 minuten te doen. Op de Weekendschool zijn de leerlingen 11 jaar en we hebben 2x75 minuten. Naar aanleiding van de eerste keer hebben we besloten om de les uit te breiden met extra concepten en niet een van de andere lessen te nemen. Deze les maakte heel wat creativiteit los en het leek beter om ze daarmee door te laten gaan.

#### 3.1 LESDOEL VAN LESDAG 1

- Leren hoe ze zonder gezondheidsklachten met beeldscherm, muis en toetsenbord kunnen werken.
- Laten kennismaken met het analyseren van een probleem.
- Eerste kennismaking met programmeren in Scratch.
- Laten zien dat je met heel weinig moeite een animatie kunt maken.
- Ze kennis laten maken met de bouwstenen van Scratch
  - o Herhalen
  - o Testen op gebeurtenissen (toetsaanslagen, het raken van een object)
  - o Beweging
  - o Variabelen
  - o Geluid
  - o Random getallen
  - o Zenden en ontvangen van berichten tussen de sprites
- Ze laten tekenen

## 4. ALGEMENE AANWIJZINGEN

### 4.1 KOPIËREN TUSSEN DE SPRITES

Je kunt opdrachten kopiëren van de ene naar de andere sprite door de opdracht of het blok opdrachten op te pakken en naar de betreffende sprite te slepen. Je zult zien dat er een lijntje om de sprite komt als die geselecteerd is. Het is het handigste om het te kopiëren blok links boven op te pakken.

Je kunt geen opdrachten naar een nieuwe sprite *verplaatsen*. Je moet kopiëren en dan de originele opdrachten verwijderen.

### 4.2 OPSLAAN

Laat ze regelmatig hun werk opslaan. Anders zijn ze het kwijt bij stroomuitval. Het is ook handig dat ze bestanden nummeren.

### 4.3 AFTEKENEN ALS MET AAN DE BEGELEIDER HEBBEN UITGELEGD

De opdrachten staan in groene blokken. Als er wordt gevraagd om de begeleider uit te leggen wat hoe iets werkt, dan staat die aanwijzing in het rood. Afhankelijk wat we als begeleiders afspreken kunnen we er voor kiezen dat je een paraaf zet als dat overleg daadwerkelijk gebeurd is. Doordat het in het rood staat, kun je heel makkelijk zien welke opdrachten het betreft.

### 4.4 VOORTGANG NOTEREN

Nu we deze les nog maar een paar keer gedaan hebben en steeds ook iets hebben veranderd willen we uitzoeken hoeveel tijd ze nodig hebben om de les te doen. Zodat we in latere lessen bijvoorbeeld weten dat ze na een uur bij opdracht X moeten zijn. Daarom krijgt elke begeleider een formulier voor de groepjes die hij of zij begeleidt en vult daar op gezette tijden (bijvoorbeeld elk kwartier) in bij welke opdrachtnummer elk groepje is. Als de leerlingen dat zien, leg ze dan uit dat het geen wedstrijd is, dus dat ze gewoon moeten doorwerken maar dat het er niet om gaat wie het snelste is. We noteren geen namen van de leerlingen. Het is puur voor ons eigen inzicht in het lesverloop.

## 5. OPMERKINGEN PER ONDERDEEL

### 5.1 HET ANALYSEREN VAN DE OPDRACHT

Vraag ze wat ze gezien hebben in het filmpje. Wat voor grote stappen of grote blokken zien ze daarin en laat die opschrijven. Eerst ieder kind voor zich, laat ze daarna per tweetal die straks gaan samenwerken bij dezelfde computer het resultaat opschrijven.

Grote blokken waar je aan kunt denken:

1. Een doolhof verkrijgen
2. Een poppetje (de kat) in vier richtingen laten bewegen aan de hand van toetsaanslagen.
3. Als de kat de muur raakt, keer dan terug naar *af* en zeg even *Au!*

4. Als de kat bij de prinses komt dan moet de kat iets uitroepen.
5. Het spel eindigen als de tijd over is
6. Het spel eindigen als de prinses is gered
7. Kopstoten kunnen tellen.
8. De prinses moet weten als de kat in gevaar is en dan op en neer springen.
9. Geluid laten afspelen

Vertel ze dan dat ze in de les nu een vast schema volgen, want we kunnen de instructies niet aanpassen aan de hand van wat ze hier bedenken. En ze moeten de basis stappen nog leren. Maar vraag ze waar ze denken dat je in de praktijk mee zou beginnen.

Antwoord: met de meest onzekere of riskante dingen, die wel essentieel zijn. Daar maak je een prototype van om te kijken of het concept werkt.

Verder is de ontwikkelmethode zo dat je iets kleins bouwt, dat gaat testen en laat zien aan de gebruikers en dan bouw je nieuwe dingen erbij. Dit is het *agile* ontwikkelmodel.

## 5.2 KIES HET DOOLHOF

Ze zouden het doolhof ook zelf kunnen tekenen, echter naar aanleiding van eerdere ervaringen hebben we het zelf tekenen vervangen door het kiezen uit een aantal verschillende doolhoven. Het zelf tekenen is naar het eind verplaatst omdat ze zich compleet kunnen verliezen in het tekenen van het mooist denkbare doolhof – dat dan soms ook nog eens niet echt als doolhof bruikbaar is – en verdwijnt als je op Clear drukt terwijl je niet precies weet wat dat doet (alles weggooien).

## 5.3 HOOFDSTUK 6: WAAROM STAAT ER EEN ZEG MET BLANKO TEKST

Er wordt gevraagd waarom er een blokje *zeg* staat met een blanco tekst. Dat is om de eerder vertoonde tekst weg te halen.

## 5.4 VARIABELEN

We laten ze de variabelen globaal aanmaken, dus beschikbaar in alle sprites. Het is in de praktijk beter om de variabelen alleen te laten gelden voor de sprite die ze nodig heeft, maar dat is een extra concept waarvan ze het nut nu niet zullen inzien.

## 5.5 SPEELTIJD

Als je test of de overgebleven speeltijd precies nul is dan zul je merken dat de teller gewoon langs de nul loopt. Dat komt omdat de tijd die je ziet waarschijnlijk een afronding is van een meer precieze tijd. En de test gebeurt vast op een moment dat de tijd niet precies nul is. Daarom moet je testen of de resterende tijd kleiner of gelijk is aan nul. In Scratch kun je niet makkelijk testen op kleiner of gelijk, wel op kleiner. Omdat dit slechts een fractie van een seconde uitmaakt, kun je gewoon testen op kleiner. Je kunt de test veranderen door rechts te klikken op het blok met het = teken en dan kleiner of groter kiezen. Echter, bedenk dat ze nog niet altijd op school de tekens < en > gehad hebben, dus dat zal je soms moeten uitleggen.

## 5.6 SIGNALLEN

Als je bij het signaal van de kat niet zou wachten na het zenden van het signaal dan worden er heel veel signalen gestuurd en dat geeft wat bijzondere effecten. Daarom wachten we na het zenden van een signaal.

## 5.7 TEKENEN VAN HET DOOLHOF

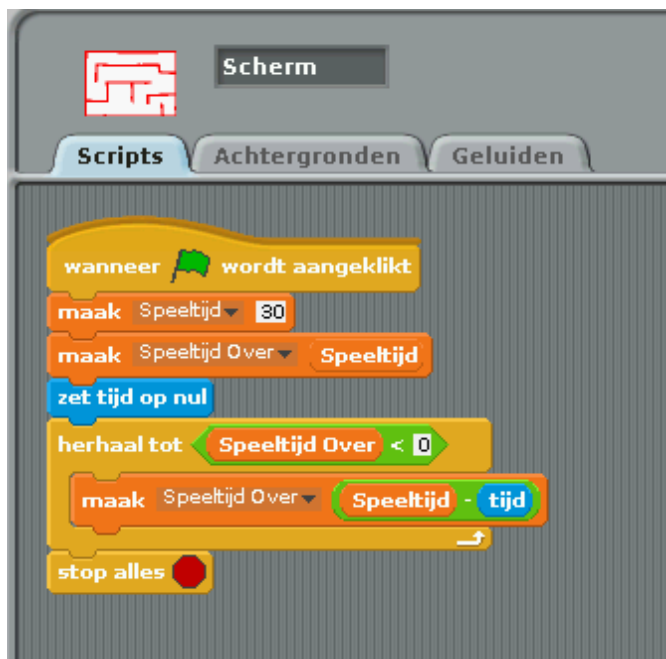
Dit onderdeel is na ervaring met eerdere lessen naar het eind verplaatst. Zoals eerder al gezegd, als je ze laat beginnen een doolhof zelf te laten tekenen dan spenderen ze daar heel veel tijd aan en komen soms met iets dat niet echt als doolhof bruikbaar is.

Als ze in verschillende kleuren tekenen, ook al lijken soms b.v. twee kleuren rood hetzelfde, betekent dat overal waar ze testen op het raken van een kleur moeten testen op twee of meer kleuren. Dat maakt het niet makkelijker. Beter om dat te vermijden.

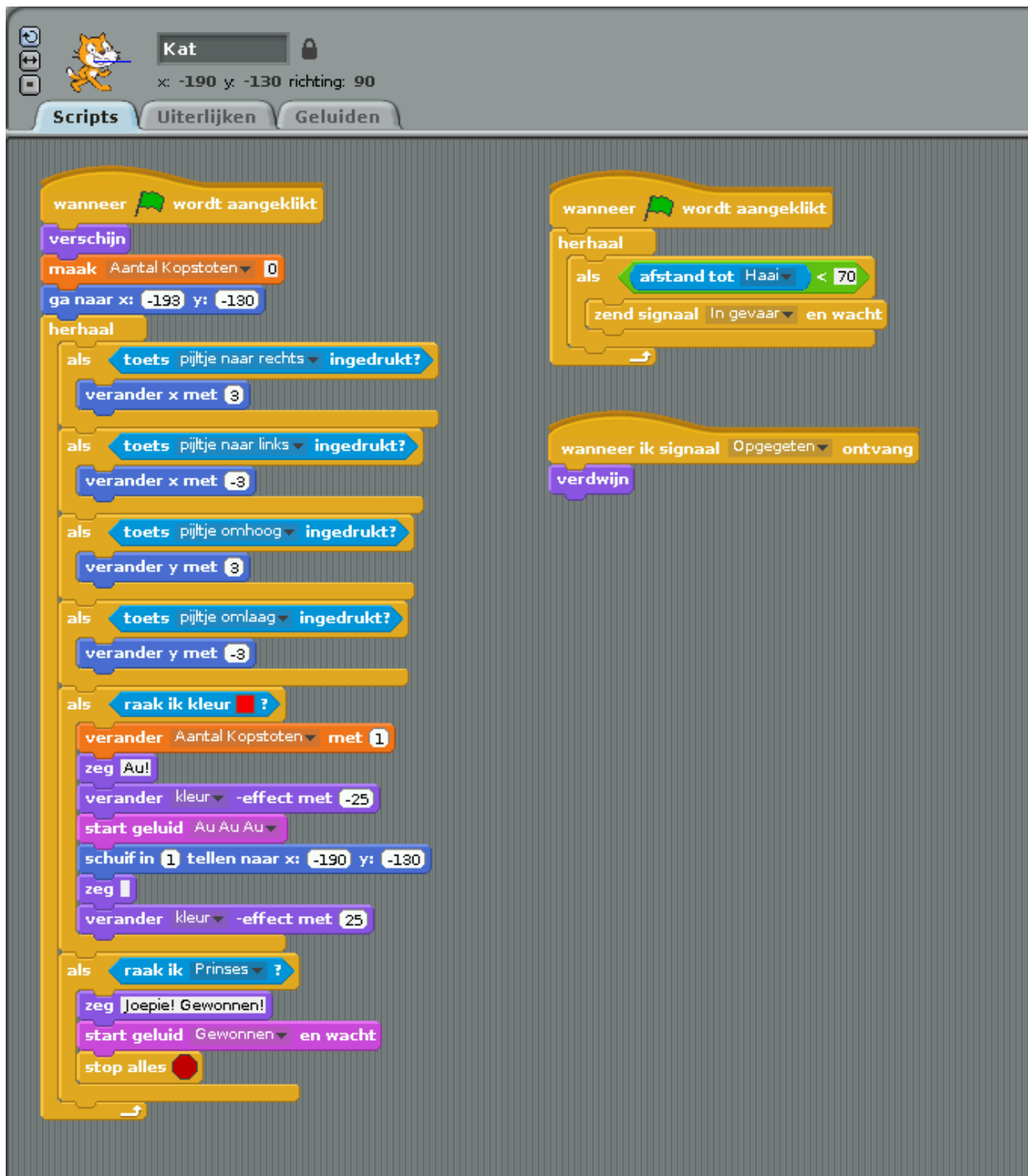
## 6. HET COMPLETE PROGRAMMA

Hieronder staat het complete programma. Drie sprites en het scherm.

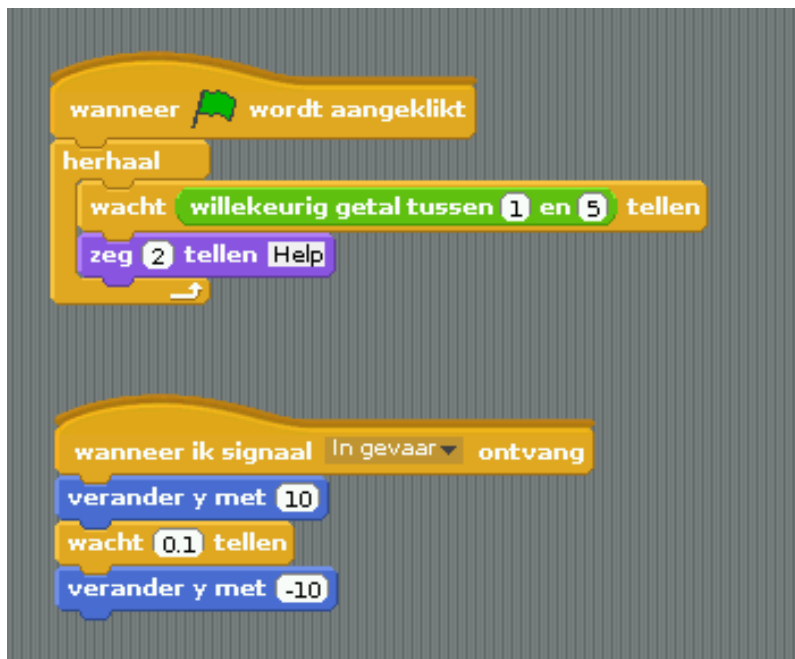
### 6.1 SCHERM



## 6.2 KAT



### 6.3 PRINSES



### 6.4 HAAI



## 7. VRAGEN EN ANTWOORDEN

Hier een aantal vragen die ze mogelijk stellen en de antwoorden

- Waarom heet dit ding Raspberry Pi?
  - o Het beestje moet een naam hebben.
  - o De bedenkers wilden een naam van fruit gebruiken om nostalgische redenen. Raspberry is een framboos. Vroeger werden meer computers naar fruit genoemd: Apricot (abrikoos) en je kent er vast nog wel een: Apple.
  - o Pi komt van Python, de naam van een programmeertaal die je kunt gebruiken. En als je Pi spelt als Pie dan betekent het: taart. Dus frambozen taart.
  - o Het heeft navolging gekregen, want iemand is Banana Pi gestart.
  - o En er is nu ook een Orange Pi, wat een complete computer is ala de Raspberry Pi, maar dan voor rond de 10 euro (excl. SD kaart, voeding, monitor, toetsenbord en muis).
- Wat kost een Raspberry Pi en deze opstelling?
  - o Deze Raspberry Pi, als je die in Eindhoven koopt kost (sossolutions.nl) 58,95 euro (De Raspberry Pi, behuizing, SD kaart en netvoeding. Gratis verzending.
  - o Tweede hands monitoren kun je kopen voor ca. 25 euro, deze waren tussen de 8,50 en 10 euro. Muizen 1,25 euro en toetsenbord 2,50 euro. Kabel voor de monitor kost een paar euro. Maar het hangt heel erg af waar iets toevallig aangeboden wordt en er kunnen verzendkosten bijkomen.
  - o Om alles nieuw te kopen moet je naast de Raspberry Pi rekenen op ca. 120 euro (100 euro monitor, 15 euro muis en toetsenbord, 5 euro HDMI kabel).
  - o Overigens zijn er ook mogelijkheden om een heel klein scherm aan te sluiten en dan ben je minder kwijt, maar heb je ook minder resolutie.
- Zijn er nog goedkopere computers?
  - o Ja. Afgezien van de Orange Pi die ik al wel heb maar nog geen tijd voor gehad om uit te proberen zullen we in de tweede en derde les een computer gebruiken (Arduino Nano) die samen met de voeding en wat cijferdisplays en knopjes ca. 5 euro kost. Maar daar kun je dan ook veel minder mee, en je hebt een grotere computer nodig (zoals de Raspberry Pi) om die te programmeren.
- Kunnen ze thuis op een computer dit ook doen?
  - o Ja, er is een beschrijving hoe ze dat moeten doen. Als ze hun naam invullen in het speciale programma en het vinkje zetten dat ze het willen hebben dan krijgen ze hun werk en dat blad met uitleg per email toegestuurd.