

# WEEKENDSCHOOL – PROGRAMMEREN –

## LES 2A – BEGELEIDERSINSTRUCTIE

### 1. INTRODUCTIE

Dit is de beschrijving van programmeren les 2A. Het geeft achtergrondinformatie, maar ook de uitgewerkte stappen.

### 2. VOORBEREIDING VOOR DE LES

Het is belangrijk om voor de dag zelf jezelf voor te bereiden.

#### 2.1 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.

#### 2.2 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 programmeren we ook in Scratch 2.0 maar dan op een Raspberry Pi. Daar zijn wat fouten waar we omheen werken, maar verder is het hetzelfde.

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. Als je dat wel wilt, vraag dan even om de geluidsbestanden.

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. Voor deze les 2A heb je ook minstens een Arduino Nano nodig. Die kan ik je sturen.

Als je wilt kun je ook een complete setup met Raspberry Pi, monitor, toetsenbord, plankje met eend etc. lenen.

## 2.3 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.

## 3. VOOR DE LES

### 3.1 OPBOUW VAN DE WERKPLEKKEN

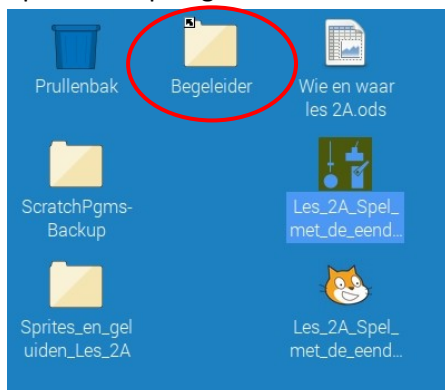
We proberen het zo te plannen dat er per begeleider maximaal 8 leerlingen zijn. Die zitten op 4 werkplekken. We proberen meer begeleiders te hebben. Elke begeleider bouwt dus maximaal 4 werkplekken op. Er is een aparte handleiding hoe dat moet.

### 3.2 KLAAR ZETTEN VAN DE RASPBERRY PI

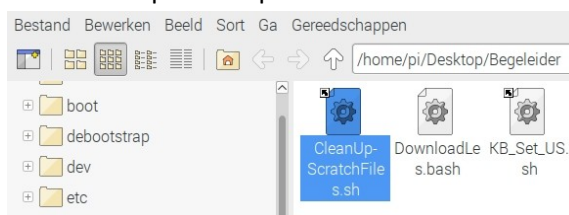
Als de werkplek klaar is, start dan de Raspberry Pi op. Daarna moeten er nog wat handelingen gedaan worden.

#### 3.2.1 START CLEANUP-SCRATCHFILES.SH

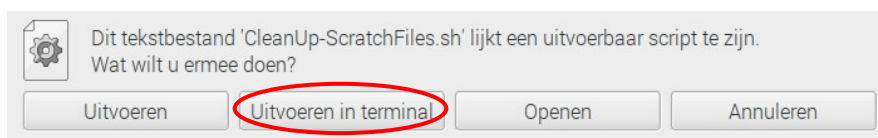
1. Open de map Begeleider



2. Dubbelklik op CleanUp-ScratchFiles.sh



3. Kies *Uitvoeren in terminal*



4. Je ziet nu dit venster, maar het kan ook zijn dat het venster vanzelf geminimaliseerd wordt. Als het zichtbaar is, minimaliseer het dan. **Het moet de hele les blijven draaien.**

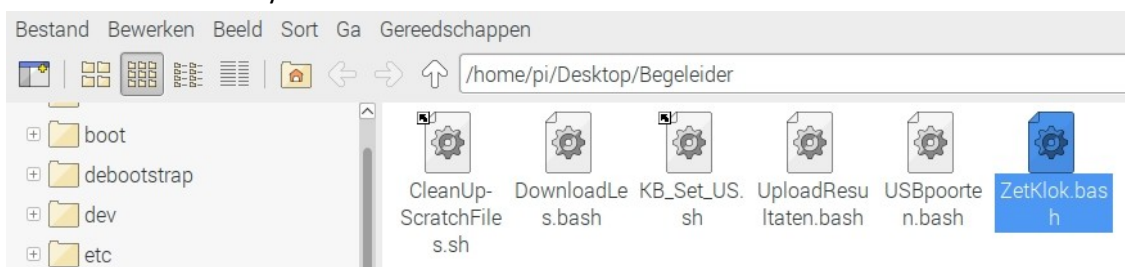
```
Bestand Bewerken Tabbladen Hulp
ThisFileName=CleanUp-ScratchFiles.sh
Now looping forever over /home/pi and moving all .sbx or .sb2 files as
.sb2 files to /home/pi/Desktop.
Any spaces in the filename will be removed.
A file of the same name - if present there - will be moved to
/home/pi/Desktop/ScratchPgms-Backup with a timestamp appended.

Good: There are running 0 instances of Scratch (1 or 2).
```

Dit programma zorgt dat als je een .sb2 of .sbx file opslaat in /home/pi deze weer als – altijd – een .sb2 file op het bureaublad wordt gezet, met de eventuele spaties verwijderd uit de filenaam. Dit is nodig wegens fouten in Scratch 2 op Raspberry Pi.

### 3.2.2 ZET DATUM EN TIJD

#### 1. Dubbelklik ZetKlokEnSysteem



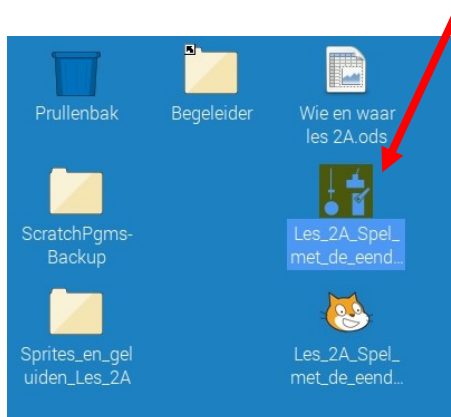
#### 2. Kies *Uitvoeren in terminal*



### 3.2.3 ZORG DAT SCRATCHCLIENT IS OPGESTART

**Let op:** deze stappen doen we pas als we met les 2A beginnen. Dus in de ochtend na het opzetten of in de lunchpauze als we les 2A pas in de middag doen.

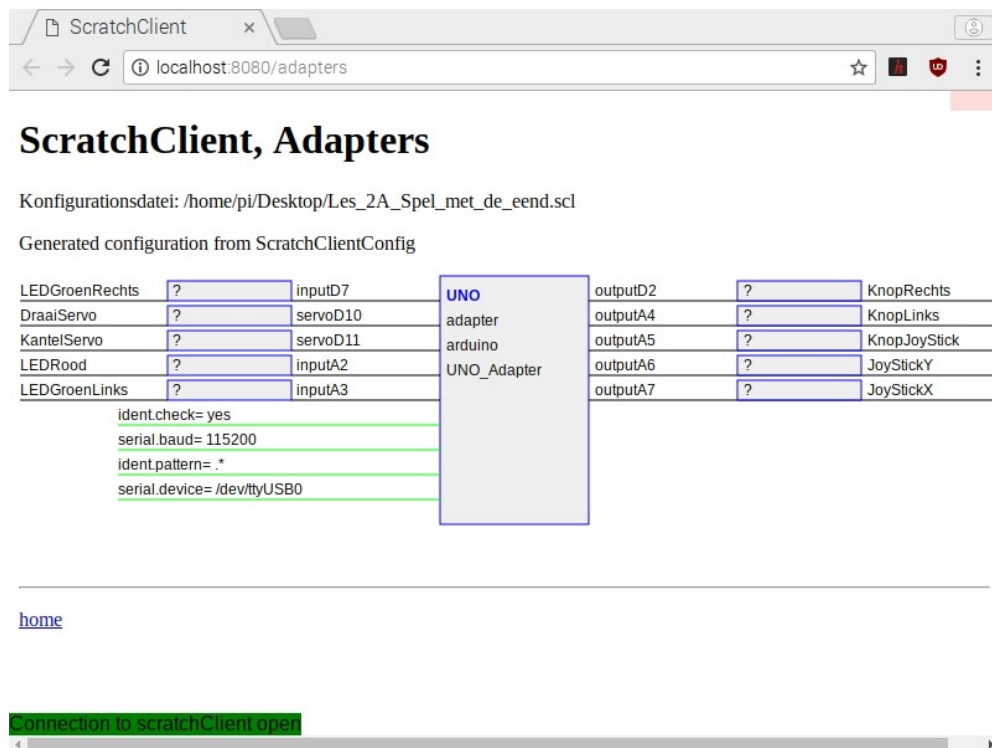
ScratchClient maakt de verbinding tussen Scratch en de Arduino op het plankje met de eend. Dubbelklik daarvoor het scratchClient icoon.



Je krijgt dan dit venster:

```
Bestand  Bewerken  Tabbladen  Hulp
<parameter name="serial.device" value="/dev/ttyUSB0"/>
/dev/ttyUSB0
0
scratchClient gets started on /dev/ttyUSB0
scratchClient uses config file: /home/pi/Desktop/Les_2A_Spel_met_de_eend.scl
scratchClient 2018-02-25
Process of scratchClient = 1886
2018-09-18 13:23:39,796 [WARNING] __main__ - 1261: There was an error connecting
to Scratch 1.4 !
2018-09-18 13:23:39,797 [WARNING] __main__ - 1263:  Unterstuetzung fuer Netzwer
ksensoren einschalten!
2018-09-18 13:23:39,797 [WARNING] __main__ - 1264:  Activate remote sensor conn
ections!
Process of the finally running instance of scratchClient = 1886
--disable-quit --enable-tcp-fast-open --disable-gpu-compositing --ppapi-flash-p
ath=/usr/lib/chromium-browser/libpepflashplayer.so --ppapi-flash-args=enable_sta
gevideo_auto=0 --ppapi-flash-version=
Fontconfig warning: "/etc/fonts/fonts.conf", line 160: blank doesn't take any ef
fect anymore. please remove it from your fonts.conf
[1900:1900:0918/132343.955279:ERROR:gpu_process_transport_factory.cc(1029)] Lost
UI shared context.
Er is een nieuw venster gemaakt in de bestaande browsersessie.
2018-09-18 13:23:44,673 [WARNING] tornado.access - 2063: 404 GET /css/style.css
(::1) 4.06ms
2018-09-18 13:23:44,695 [WARNING] tornado.access - 2063: 404 GET /icon/16x16_Del
ete.jpg (:1) 6.50ms
2018-09-18 13:27:00,372 [WARNING] __main__ - 1261: There was an error connecting to Scrat
ch 1.4 !
2018-09-18 13:27:00,373 [WARNING] __main__ - 1263:  Unterstuetzung fuer Netzwerkse
nsoren einschalten!
2018-09-18 13:27:00,374 [WARNING] __main__ - 1264:  Activate remote sensor connections!
```

Ook wordt een browser gestart. Daarin kun je de variabelen zien zodra die een waarde hebben gekregen. Ook die kun je het beste minimaliseren.



#### 4. GEZOND COMPUTEREN

Als we het vorige week nog niet gedaan hebben dan geven we een korte introductie over hoe je gezond moet computeren. Als we dat wel gedaan hebben, dan vragen we ze om de 5 regels uit het filmpje op te noemen.

- We beginnen met een introductie over hoe ze moeten computeren zonder gezondheidsklachten te krijgen. Dit gebeurt door de gastdocent. Er is een aparte instructie. Begeleiders hebben geen functie in dit onderdeel.
- Daarna 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.

#### 5. WAT ZE GAAN MAKEN

- We laten een filmpje zien over wat ze deze les gaan maken. **Let op:** Als we les 2A in de middag beginnen dan spelen we dit filmpje ook pas in de middag af.
- Dit is de Scratch 1.4 versie. De Scratch 2.0 versie moet nog opgenomen worden.  
<https://www.youtube.com/watch?v=Qo1gnXNzhqE>
- 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 (een aantal tweetallen) met een begeleider zitten en praten over welke poppetjes er zijn en wat die doen.
- De begeleider zit hierbij samen met een aantal tweetallen. Als je klaar bent met de discussie dan ga je naar de computers.
- Dan beginnen we met de opgave *Les\_2A\_Spel\_met\_de\_eend*

#### 6. WELK LEERPROCES?

De Lifelong Kindergarten group van het MIT in Boston, waar Scratch ontwikkeld is, propageert het Creative Learning: leren door te doen. Zij leggen de nadruk op het door de leerlingen zelf laten ontdekken en spelen en zo te leren. Ik geloof daar ook zeer in, echter we hebben maar drie zondagen en willen ze toch wat meer meegeven. Daarom is gekozen om ze opdrachten te laten maken. Voor spelen en eigen inbreng is gelegenheid aan het eind. Maar als ze eerder zelf dingen willen uitproberen en een wat andere richting in gaan dan is dat prima, maar kijk wel dat ze niet in een onderwerp blijven hangen omdat ze de rest (zogenaamd) niet snappen.

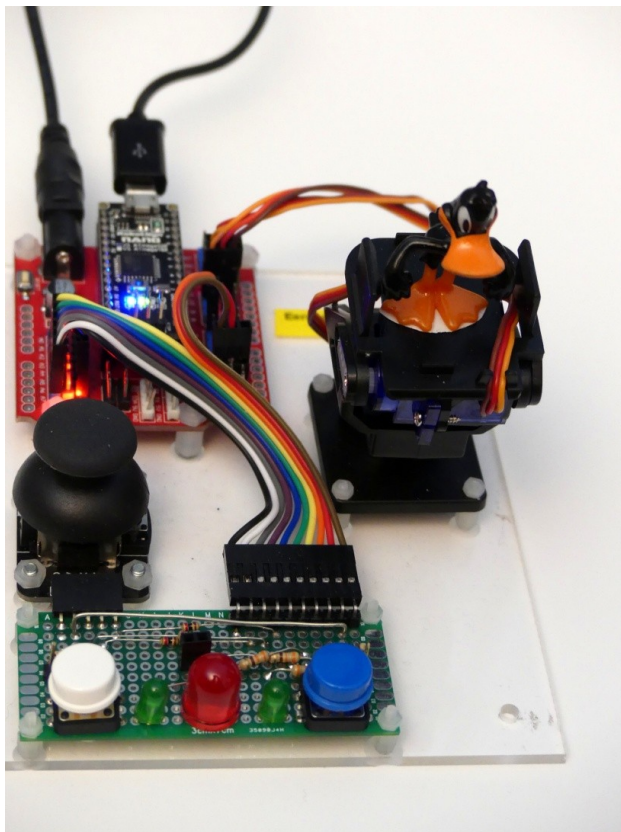
Als ze andere sprites willen gebruiken, dan mag dat. Maar maak ze duidelijk dat als ze een sprite weggooien ook alle scripts, uiterlijken en geluiden worden weggegooid. Laat ze dan eerste de nieuwe sprite importeren, dan de scripts overzetten naar de nieuwe sprite en pas nadat ze hebben gecontroleerd dat het gelukt is de oude sprite weggooien. Laat ze ook altijd eerst hun werk opslaan.

## 7. LESDOEL VAN LESDAG 2

- Leren hoe ze zonder gezondheidsklachten met beeldscherm, muis en toetsenbord kunnen werken.
- Laten kennismaken met het analyseren van een probleem.
- Verdere 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 leren dat programmeren niet alleen gaat over het veranderen van pixels op een scherm, maar dat je ook LEDjes, motortjes, schakelaars etc. kunt aansturen (physical computing).

## 8. DE OPSTELLING

We gebruiken een Raspberry Pi, monitor, toetsenbord en twee muizen, plus een bordje met knopjes en lampjes en een eend op een draai/kantel plateau. Op dat bordjes zit ook een Arduino Nano die alles bestuurt en communiceert via USB met een programma *scratchClient* op de Raspberry Pi.



## 9. VERBINDING MET DE ARDUINO MAKEN

Volg de leerlingenhandleiding om Scratch te starten (dubbelklik) en de verbinding met scratchClient te leggen (druk op de groene stip). De LED op de Arduino moet nu na een paar seconden langzaam gaan knipperen. Dan is de verbinding gelegd.

### 9.1 WAT KAN ER FOUT GAAN?

#### 9.1.1 EXTENSIEBLOKKEN ZIJN NIET ZICHTBAAR OF DE DATA KOMT NIET AAN

Als de extensieblokken niet zichtbaar zijn in Scratch terwijl scratchClient wel loopt en contact lijkt te hebben (als je op een knop op het bord drukt dan zie je iets in het venster van scratchClient veranderen), dan kan het zijn dat er twee Scratch programma's lopen. Stop degene die je niet nodig hebt.

Je zult in dit geval ook zien dat het programma *CleanUp-ScratchFiles.sh* in de statusbalk knippert.



## 10. DE UITWERKINGEN VAN DE OPDRACHTEN

**OPDRACHT 13:** De linker groene LED laten branden zolang er wordt afgeteld.

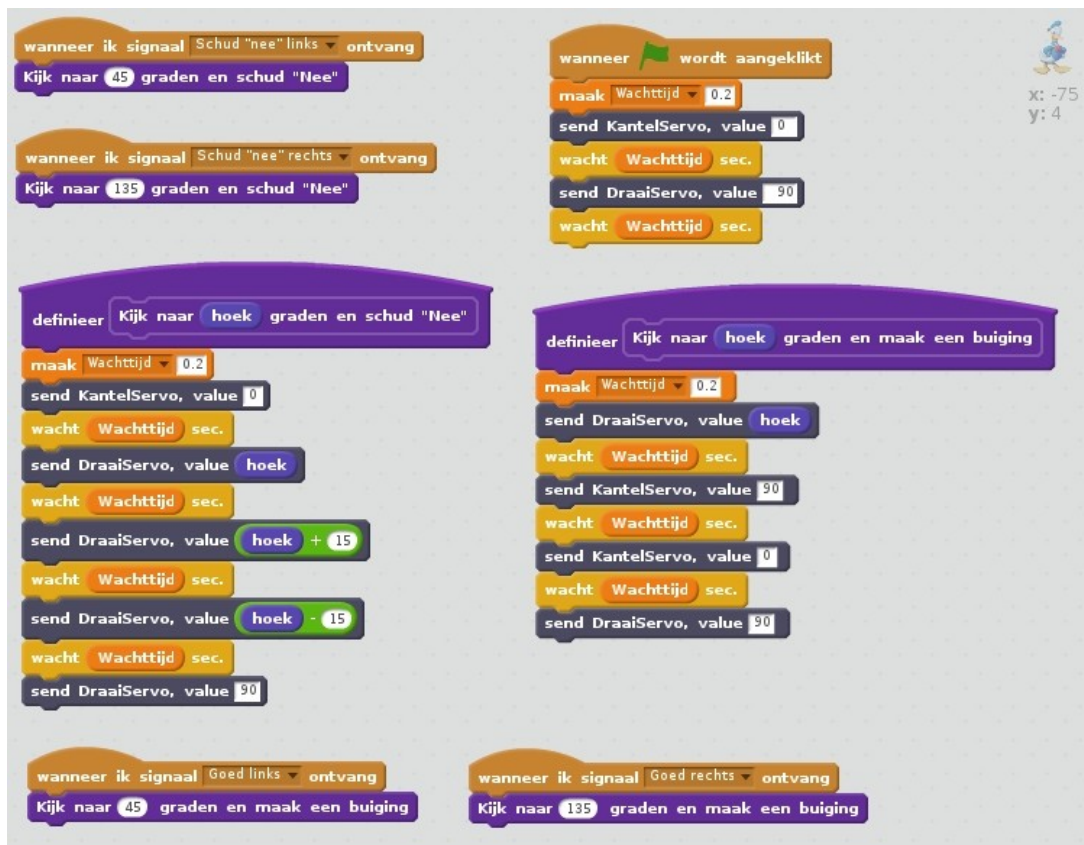


**OPDRACHT 14:** De rechter groene led laten branden gedurende de laatste 5 seconden van het aftellen





## OPDRACHT 26: Buigen van de eend



## 11. HET ECHTE WERK

### OPDRACHT 29: Laat de rode LED knipperen



### OPDRACHT 31: De lus voor de knoppen



## OPDRACHT 32: Rechter knop behandelen



## OPDRACHT 33: Linker knop behandelen



## OPDRACHT 34: Afhandeling van gelijkspel

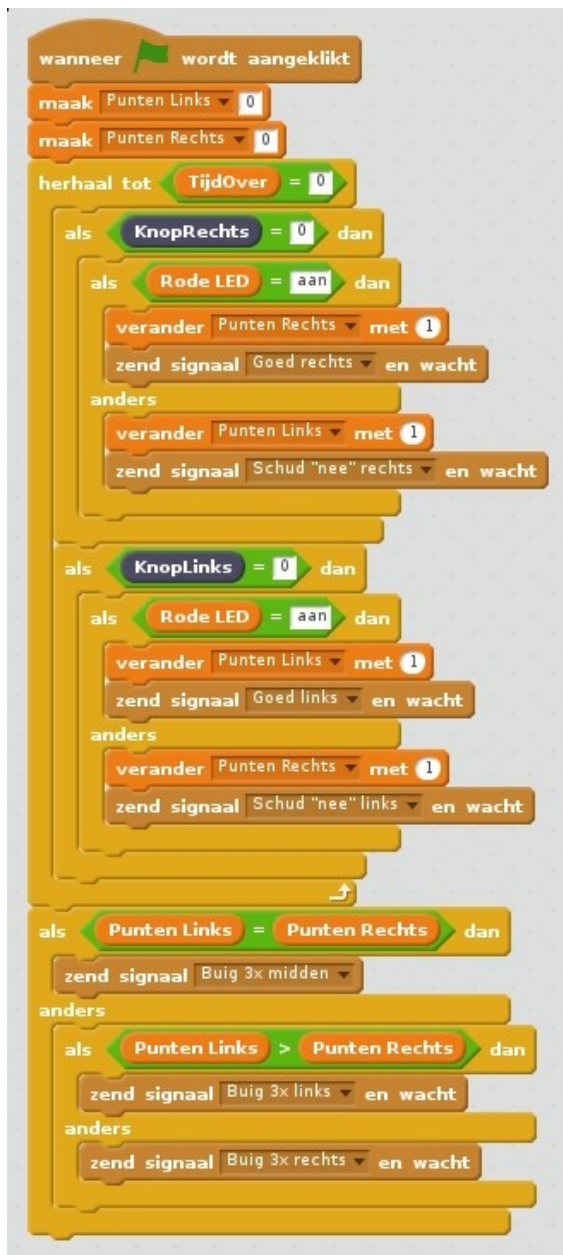


In de speelveld sprite



In de WS eend sprite

## OPDRACHT 35: Geen gelijkspel



In de speelveld sprite



In de WS Eind sprite

Deze opdrachten gaan goed, maar er zit wel een beetje een trucje in. De opdracht *Kijk naar 135 graden* en *Kijk naar 45 graden* eindigt met het zetten van de draaiservo in de 90 graden positie. Als je er een paar achter elkaar geeft zou je verwachten dat de eend steeds naar het midden teruggaat. Echter, omdat er niet gewacht wordt na het naar het midden bewegen van de draaiservo gaat draait de eend niet terug naar het midden, omdat de volgende opdracht voor het kijken naar 45 of 135 graden zo snel komt dat het draaien naar 90 graden niet meer gebeurt. Je ziet de eend nog wel aanstalten maken, maar heel snel weer in 45 of 135 graden stand komen.

Dit is normaal het punt waar je zou moeten besluiten om het blok anders te laten eindigen. Maar voor de les nemen we het zoals het is.



### OPDRACHT 39: Verander de uiterlijken van de eend



### OPDRACHT 40: Verander de uiterlijken van de rode LED sprite



### OPDRACHT 41: Geluid



### OPDRACHT 43: Maak het zo dat je het thuis ook kunt spelen

Ze moeten een test toevoegen bij het testen of een knop is ingedrukt om ook te kijken of een toets op het toetsenbord is ingedrukt.

### OPDRACHT 44: Spelronde begint als je op de joystick knop drukt

Nu start het spel als je op de groene vlag drukt. Ze moeten een lus maken die eindeloos loopt en telkens wacht op de overgang van Joystickknop niet ingedrukt (waarde 1) naar ingedrukt (waarde 0). Dan moet er een bericht gestuurd (b.v. start spel). Die moet worden opgevangen door de klok, de rode LED sprite om te starten en de speelveld sprite. De speelveld sprite test of de klok op 0 staat, dus kijk wel uit voor race condities, als de klok het bericht later verwerkt dan dat de speelveld sprite test of de tijd over is. Mogelijk moet je een extra signaal Reset maken om de puntentelling op nul te zetten en de tijd op 15 seconden.

#### **OPDRACHT 45: Is het spel eerlijk?**

Het spel kan natuurlijk op veel manieren bewust oneerlijk gemaakt worden. Bijvoorbeeld door een tijdvertraging in te bouwen in de wachtlus die naar de knopjes kijkt, zal het knopje dat als eerste wordt getest na de tijdvertraging een grotere kans hebben om te worden gedetecteerd als winnaar.

Bijvoorbeeld, als een speler op het knopje drukt, dan luistert de speelveld sprite pas weer als het signaal naar de eend is verwerkt. En dat duurt even. Drukt de andere speler in die tijd op de knop, dan wordt die niet gezien. Om dat te voorkomen is er een heel andere opzet nodig.

Maar er kunnen ook onbewust elementen van onbalans in zo'n systeem zitten. En een vraag die ze zichzelf zouden moeten stellen is: wat gebeurt er als beide knopjes tegelijk worden ingedrukt? Wie wint er dan?

En hoe zou je dat kunnen testen? Hoe kun je de knopjes exact tegelijk indrukken? Dat doen we door beide knopjes aan elkaar te verbinden met een draadje. Als je dan op 1 knopje drukt, activeer je beide signalen.

Dus laat ze het draadje bevestigen. Dan is het ook handig om de random tijd voor het rode LEDje te vervangen door een constante tijd van b.v. 2 seconde, of minder als je ook stopt om de eend te laten bewegen.

Laat ze kijken wat er gebeurt. Het blijkt dat het programma niet eerlijk is. Ik heb nog niet gevonden waar het precies aan ligt.

Vraag ze om te bedenken hoe vaak ze het zouden moeten testen om betrouwbaar te zijn. En vertel ze dat er wiskunde bestaat (kansberekening) om dat goed te kunnen bepalen. Dat krijgen ze op de middelbare school.

En vraag ze ook: "Als nu steeds de ene wint, wil dat zeggen dat het spel echt oneerlijk is?".

Leg ze voor wat er zou gebeuren als het zo was dat als het linkerknopje 1 ms (test eerst even of ze weten dat milli staat voor 1/1000) later wordt ingedrukt de kans op winnen wel 50% is, terwijl als ze echt tegelijk worden ingedrukt het linkerknopje steeds wint. Het spel is dan theoretisch oneerlijk, maar is het in praktijk echt erg? In het verkeer is je reactietijd 1 seconde. In het geval dat je geconcentreerd bent op een LEDje en knopje is dat veel minder. Internet geeft 0,2 seconde aan. Bij hardlopen geldt een reactietijd van minder dan 0,1 seconde als valse start, omdat je dan geacht wordt al gestart te zijn voordat je op het startschot had kunnen reageren.

Dus zou 1 ms vertraging op een 200 ms reactietijd (die ook bij iedereen wat anders is) erg zijn?

Dus als het ene knopje een grotere kans heeft om te winnen, hoe zou je dan je test kunnen aanpassen om te kijken hoe (on)eerlijk het is? Antwoord: zet een elektrische vertraging tussen het ene en het andere knopje. Dat kunnen we hier niet doen (geen materiaal).