

De Astro Pi programmeren

Mission Zero

De Astro Pi is een kleine computer aan boord van het internationale ruimtestation (ISS). En jij kunt deze vanaf jouw computer programmeren! In deze les leer je de belangrijkste commando's voor de Astro Pi. Na afloop van deze les kun jij meedoen met Mission Zero!

Wat heb je nodig?

- een pc met internet



De temperatuursensor

De Sense HAT heeft een temperatuursensor waarmee je de temperatuur kunt meten. Schrijf onderstaande code over en voer uit.

```
from sense_hat import SenseHat
sh = SenseHat()

#Lees de temperatuur van de sensor en sla op als temp
temp = sh.get_temperature()

#rond de temperatuur af op twee decimalen
temp = round(temp,2)

#zet het nummer om in een stuk tekst
temp = str(temp)

sh.show_message("temp")
sh.show_message(temp)
```

Stel jezelf de volgende vragen:

Wat gebeurt er als je de regel met round weglaat?

Wat doet de functie str?

Wat is het verschil tussen "temp" en temp?

Meer experimenteren

Probeer ook eens de volgende regel uit:

```
sh.show_message(str(round(sh.get_temperature())))
```

Op deze manier voer je alle opdrachten in één regel uit. Vind jij dit handiger of juist niet?

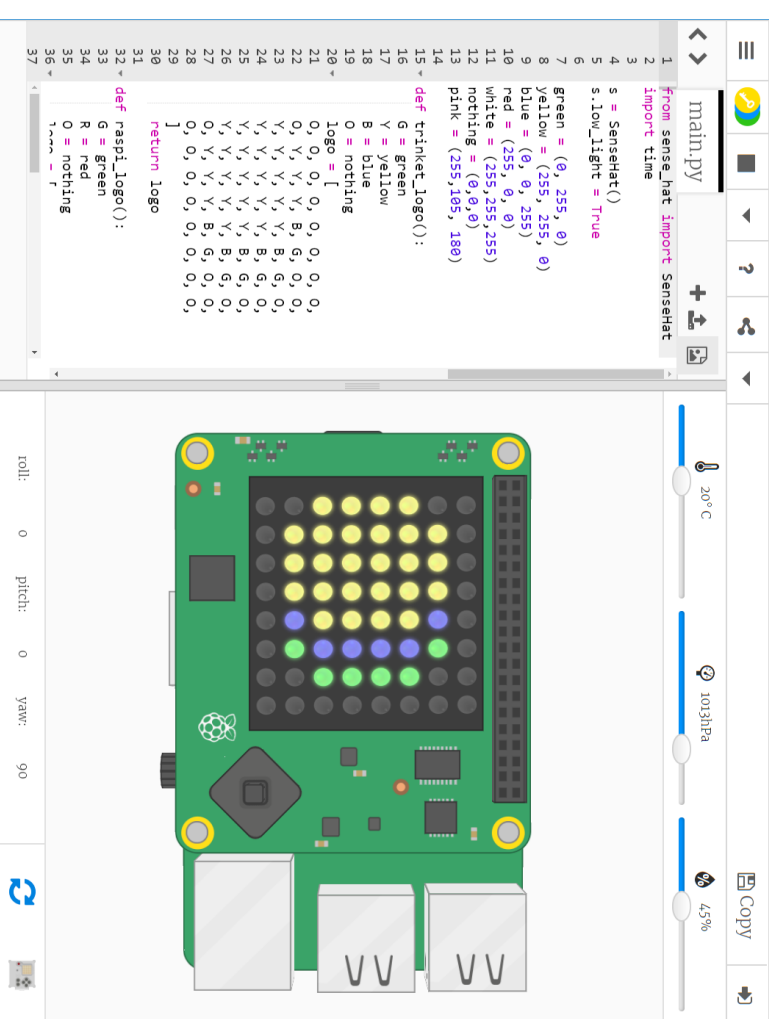
Schrijf nu een programma waarbij je elke seconde de temperatuur op het display laat zien. Maak gebruik van het `while True`:-commando. `while` is een lus, een opdracht die herhaald wordt zolang aan een bepaalde voorwaarde wordt voldaan. Daarna staat er `True`; dat betekent dat er altijd aan de voorwaarde wordt voldaan en de opdrachten oneindig vaak herhaald worden.

```
while True:
    sh.show_message(str(round(sh.get_temperature()))),
    scroll_speed=0.03)
time.sleep(1)
```

Let op! Na `while True`: staat een dubbele punt en de regels eronder zijn ingesprongen

Starten

Ga naar www.ruimtevaartindeklas.nl/astropi. Klik op 'Nu oefenen met de Astro Pi'. Klik op de link om volledig scherm weer te geven. Op het scherm zie je het onderstaande venster.



The screenshot shows a web browser window with a code editor on the left and a camera view on the right. The code editor contains Python code for a Raspberry Pi program that uses a SenseHat to display a 5x5 grid of colored LEDs. The code defines colors (green, blue, yellow, red, pink) and a function to print the 'raspi' logo using the LEDs. The camera view shows a Raspberry Pi board with a camera module attached, displaying a grid of colored LEDs on its screen. The interface includes a temperature slider set to 20°C, a volume slider set to 1013hPa, and a zoom slider set to 45%. There are also buttons for 'Copy' and a refresh icon.

In het scherm links staat een code. Deze kun je wissen en vervangen door je eigen code. Rechts zie je hoe de Astro Pi reageert op jouw code. Met de stop-en-runknop linksboven kun je het programma stoppen of laten afspelen

Je eerste computerprogramma

Schrijf in de Sense HAT-emulator de volgende code:

```
#Importeer de functies van de Sense Hat
from sense_hat import SenseHat

#maak een variabele waarmee je functies van de Sense Hat
#eenvoudig kunt oproepen
#dit is niet per se nodig, maar anders moet je overal
#waar sh staat SenseHat() invullen
sh = SenseHat()

#laat de volgende tekst op het display zien
sh.show_message("Groetjes van de aarde!")
```

sh.show_message laat tekst verschijnen op het ledscherm. Je kunt zelf bepalen wat voor tekst je invoert. Je kunt ook de scroll snelheid veranderen. Probeer maar eens het volgende commando in te voeren:

```
sh.show_message("Groetjes van de aarde!", scroll_speed=0.05)
```

Extra: plaatjes maken (1)

Een scherm bestaat uit kleine puntjes, pixels. Op het ledscherm kun je een afbeelding van 8-bij-8-pixels maken. Ontdek hier hoe je de pixels kunt aansturen.

Schrijf het volgende programma

```
from sense_hat import SenseHat
sh = SenseHat()
sh.clear()

#zet de x- en de y-coördinaat op 0
x=0
y=0

#zet de kleuren op maximale sterkte
r=255
g=255
b=255

sh.set_pixel(x,y,r,g,b)
```

Verander x in waarden 0 t/m 7. Wat gebeurt er op het scherm?

Doe hetzelfde voor y.

Waarom kun je voor x of y geen 8 of hoger invullen?

Extra: plaatjes maken (2)

Via het commando `set_pixel`s bestuur je het hele scherm in één keer. Zo hoef je niet voor elke pixel apart een commando te schrijven!

Schrijf onderstaand programma en voer uit.

```
from sense_hat import SenseHat
sh = SenseHat()
sh.clear()

w=(255 ,255 ,255)
z=(0,0,0)

smiley = [
    z,z,w,w,w,z,z,
    z,w,z,z,z,z,w,z,
    w,z,w,z,z,w,z,w,
    w,z,z,z,z,z,w,
    w,w,z,z,z,z,w,w,
    w,z,w,w,w,z,w,
    z,w,z,z,z,w,z,
    z,z,w,w,w,z,z]

sh.set_pixels(smiley)
```

Verander de afbeelding door de kleuren te veranderen.

Wat kun je nog meer tekenen?

Meer experimenteren

Probeer ook eens het volgende programma uit:

```
from sense_hat import SenseHat
sh = SenseHat()

#Importeer functies uit 'time' om de klok te gebruiken
#nodig voor time.sleep
import time

sh.show_letter("h")
time.sleep(0.3)
sh.show_letter("o")
time.sleep(0.3)
sh.show_letter("i")
```

Stel jezelf de volgende vragen:

- Wat zijn de verschillen tussen `show_message` en `show_letter`?
- Wat doet `time.sleep`?

De kleuren van het scherm

Het scherm van de Sense HAT bevat 64 leds die verschillende kleuren kunnen weergeven. In elke led bevinden zich drie kleinere leds: een rode, een groene en een blauwe. Je kunt alle drie deze leds apart aansturen, zodat je alle kleuren van de regenboog kunt maken.

Schrijf het volgende programma:

```
from sense_hat import SenseHat
sh = SenseHat()

#de drie basiskleuren van de ledlampjes
#voor elke kleur kun je een getal tussen 0 en 255
#invullen
rood = (255,0,0)
groen = (0,255,0)
blauw = (0,0,255)

sh.show_message("rood", text_colour=rood)
sh.show_message("groen", text_colour=groen)
sh.show_message("blauw", text_colour=blauw)
```

Probeer nu zelf eens kleuren te maken, net als hierboven de kleuren rood, groen en blauw zijn gemaakt.

Verder kun je het getal dat hoort bij een kleur, rechtstreeks invoeren in sh.show_message. Hieronder een voorbeeld.

```
sh.show_message("welke kleuren zijn dit?", text_
colour=(255,255,0), back_colour=(255,0,255))

#maak het scherm weer zwart
sh.clear()
```

Wat je hebt geleerd over programmeren

Gefeliciteerd! Je weet nu alle codes die je nodig hebt om Mission Zero te voltooien. Maar behalve een aantal commando's heb je ook geleerd wat programmeren is en wat je nodig hebt om een goed programma te maken.

- Een computer doet precies wat je zegt, maar heeft geen idee wat hij doet. Dat bepaal jij helemaal.
- Computers gebruiken andere informatie dan wij. Voor ons is groen een kleur, voor de computer is 'groen' een getal of een stukje tekst.
- Je moet heel precies zijn. Een kleine spelfout kan betekenen dat je programma niet meer werkt.
- Probeer te denken als een computer: als ik dit doe dan gebeurt er dat.
- Als je een programma schrijft, ben je veel aan het uitproberen, fouten maken en verbeteren.
- Als je maar lang genoeg doorzet, kun je elk probleem oplossen!

Tot slot op de achterkant de allerlaatste uitdaging!

Een echte uitdaging!

Voor deze opdracht heb je meerdere dingen nodig die je op de vorige werklbladen hebt geleerd.

Probeer de kleur van het scherm aan te passen als de temperatuur een bepaalde waarde heeft. Bijvoorbeeld blauw onder de 20 graden, groen tussen de 20 en 30 graden en rood boven de 30 graden.

Gebruik de commando's `if (temp<20)` : en `else`:. Let weer op de dubbele punt en de ingesprongen regels.

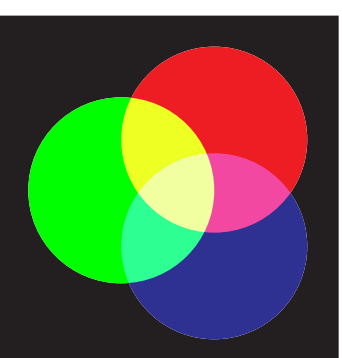
```
r = (255 ,0 ,0)
b = (0 ,0 ,255)
g = (...?....)
while True:
    temp=sh.get_temperature()
    if (temp<20):
        sh.show_message(":" , text_colour=b)
    else:
        ..
```

Bij dit programma ontbreken nog een aantal zaken:

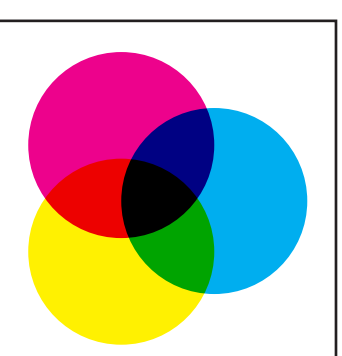
- Wat moet er onder `else`: komen te staan?
- Schrijf ook een regel voor een temperatuur boven de 30 graden.
- Welke regels moet je toevoegen om ook de temperatuur zelf weer te geven?

Meer weten over kleuren

Er bestaan oneindig veel kleuren, maar jouw ogen kunnen er maar drie zien: rood, blauw en groen. Elke andere kleur die jij ziet, is een mengsel van deze drie kleuren. Zo is geel voor ons hetzelfde als rood plus groen. Wit is voor ons hetzelfde als alle drie de kleuren samen.



Kleuren mengen met licht



Kleuren mengen met verf

Let op: kleuren mengen met licht werkt anders dan kleuren mengen met verf. De drie basiskleuren van verf zijn magenta (rozerood), cyaan (soort blauw) en geel. Dit komt doordat je met lichtbundels kleuren toevoegt en bij verf juist kleuren filtert.

Probeer zelf nieuwe kleuren te maken door de getallen te veranderen. Welke kleur vind jij het mooist?

