

## NLT Robotica

Rana Abdelrahman, Wies van Slooten, Anne Sluiter, Liv del Canho & Mette Schukinck Kool  
5e Klas Barlaeus  
13/06/2021  
Mark Kooijman  
Groep D

### Inleiding

Wij hadden allemaal nog nooit iets gedaan met programmeren en hadden dus geen idee hoe we een code moeten schrijven of hoe het er überhaupt uit zou moeten zien. Dit maakte het erg lastig om goed te beginnen en het ging bijvoorbeeld al mis bij sommigen om de programma's te downloaden. Wel waren we erg benieuwd hoe het zou lopen en wilden we leren hoe we kunnen programmeren, aangezien dit een steeds grotere rol speelt in de samenleving.

### Hoe probeerden we het aan te pakken en wat kwam eruit?

#### Downloaden van alle bestanden

Als eerst probeerden we alle software te downloaden op al onze computers. Hier liepen enkele van ons tegen problemen aan. Het downloaden van git en thonny lukte over het algemeen wel goed.

Vervolgens moesten we de tinybit op de microbit zetten via thonny, maar bij liv werkte dit niet. De library wilde alleen maar downloaden als een tekst bestand (.txt) in plaats van een hex-bestand (.hex). Hierdoor lukte het niet om de software naar thonny up te loaden en konden we de code niet op de tinybit krijgen. We kregen namelijk steeds een error bij het invoeren van de code dat de naam 'tinybit' niet bestond.

#### Schrijven van de code volgen van de lijn

Niemand van ons had ooit iets met programmeren of code schrijven gedaan. We hadden dus geen idee hoe het er überhaupt uit moest zien. Dit maakte het erg moeilijk om te beginnen. Als we hadden geweten hoe het programma werkte, was het makkelijker geweest om te beginnen. We wisten wel bijvoorbeeld wat de robot in elk geval zou moeten doen, maar we wisten niet hoe we dit aan de robot moesten vertellen. We kenden geen 'computertaal'. We hebben toen de hulp gevraagd van Mark en dit heeft ons enorm geholpen. Hij heeft uitgelegd wat voor soort 'commands' er bestonden. Ook heeft hij uitgelegd hoe we een loop moesten maken. Dit betekent dat hij een commando vaak opnieuw uitvoert in plaats van maar één keer. De term 'sleep' kenden we ook niet en heeft ons ook enorm geholpen.

We liepen wel vast toen het aankwam bij het uitkiezen van snelheden. We wisten dat de code klopte, maar het lukte de tinybit niet goed om de lijn te volgen. We hebben daarom vaak verschillende snelheden geprobeerd, waarbij ook de snelheden van het draaien en rechtdoor verschillen. We merkten dat als de snelheid 40 of minder was, dat de tinybit dan slecht of niet reed. Ook bij het draaien merkten we dat bij lage snelheden het ene wielje het beter deed dan het andere. We moesten dus 'hogere' snelheden gebruiken, maar hoge

snelheden zorgden ervoor dat de tinybit uit de bocht vloog. Uiteindelijk zijn we er wel uitgekomen met een snelheid net groot genoeg om te blijven rijden, en net klein genoeg om niet uit de bocht te vliegen.

#### Schrijven van de code obstacle avoidance

Toen we wisten hoe thorny en 'computertaal' werkten, was het een stuk eenvoudiger om de code te schrijven voor het ontwijken van een object. Dit leverde een stuk minder problemen op dan bij het volgen van een lijn. We wisten welke commando's er bestonden en welke we moesten gebruiken.

We moesten voor deze opdracht een sensor gebruiken, die nieuw voor ons was. Er stond namelijk nergens of de afstand die hij mat tot het object in centimeter, in millimeter of zelfs in kilometer was. We begonnen daarom met afstand 300. Dit bleek niet goed te lukken, dus hebben we een lager getal genomen. We probeerden 100, daarna 50, 40, 30, 20 en 10. Bij tien merkten we dat het eigenlijk erg goed lukte. We probeerden daarna ook de afstanden 1, 2 en andere kleine afstanden. We merkten toen wel dat hij het voorwerp zag, maar omdat de tinybit op snelheid was, was er geen tijd om erna een commando meer uit te voeren, zonder dat hij tegen het voorwerp aan zou botsen. We hebben 10 als uiteindelijke afstand gebruikt, omdat we hierbij merkten dat hij het voorwerp 'zag' en vervolgens nog precies genoeg tijd had om naar links te draaien.

We hadden als eerst het plan om de robot een tijdje naar links te draaien en daarna weer terug te laten draaien, zodat hij weer terug zou komen op de lijn. Dit lukte niet precies, omdat het even puzzelen was met hoelang en hoe hard de tinybit moest rijden. We hebben denken we wel 30 keer een andere snelheid of tijd geprobeerd. Soms draaide hij niet ver genoeg, of kwam hij niet meer uit bij de lijn. We hebben een waterfles neergezet op de baan om te gebruiken als object, waar de tinybit omheen moest. De breedte van het spa flesje zit ergens tussen de 5 en 7 cm.

#### extra additional features

Om te bedenken wat we als additional features wouden schrijven, hebben we rondgekeken op <https://microbit-micropython.readthedocs.io/en/v2-docs/index.html>. Toen besloten we om gebruik te maken van de random en gestures functie. Dit ging zonder problemen. Als je de robot schudde gaf hij een random getal van 1 tot en met 6, dat hij weergaf met de led lichten door middel van de display.scroll functie. de getallen 1 tot en met 6 hadden we in een rijtje geschreven en de random functie nam hier random eentje uit. Om het programma nog wat leuker te maken hebben we geschreven dat als de robot stil ligt, hij een vierkant weergeeft met de led lichten, om zo een beetje op een dobbelsteen te lijken. Ook hebben we een muziekje toegevoegd, die zich afspeelt nadat je de dobbelsteen hebt geschud. De sleep functie hebben we ook weer gebruikt om een seconde te wachten voordat het random getal wordt weergegeven. Toen was de eerste versie van dobbelsteen2.py af.

Hierna hebben we het blijenboos.py programma geschreven om het gebruik van de knopjes uit te testen en de spraak functie. Het programma geeft met de led lichten een verward gezicht weer. Als op knop a wordt gedrukt, geeft de robot een blij gezicht weer en zegt de robot 'ik ben blij'. Als de b knop wordt ingedrukt geven de led lichten een verdrietig gezicht weer en zegt de robot 'ik ben boos'. Het programma schrijven ging goed, maar de spraakfunctie werkte slecht en was onverstaanbaar. We hebben daarom meerder speeds

uitgetest om te kijken of dat het verstaanbaarder maakte. Helaas was dit niet het geval. We hebben daarom besloten om de spraakfunctie niet nog een keer te gebruiken, maar de knoppen wel.

In het programma dobbelsteen2.py hebben we de knoppenfunctie toegevoegd. We hebben een nieuwe rij met antwoorden toegevoegd, waaruit de random functie zijn antwoorden neemt. Als knop a wordt ingedrukt geven de led lichten nu een 8-ball antwoord weer met de display.scroll functie. Ook hierbij hebben we de sleep functie gebruikt en voor de leuk een muziekje toegevoegd. Als de b knop wordt ingedrukt geeft de robot een korte uitleg van mogelijkheden door met de display.scroll functie “schud om een dobbelsteen te gooien en klik linker knop voor een antwoord op je vraag!” weer te geven op de led lichten. Ook dit ging allemaal zonder problemen.

Hierna hebben we opnieuw rondgekeken op <https://microbit-micropython.readthedocs.io/en/v2-docs/index.html> om nog een andere extra feature toe te voegen. Eerst wilden we gebruik maken van de g-force gesture functies. Maar deze leken niet te werken. Ook hebben we geprobeerd de kompasfunctie te gebruiken. Deze werkte wel, maar hier wisten we niks om aan toe te voegen. Daarom hebben we besloten om nog een leuke simpele opening voor onze groep te schrijven, waarbij onze naam: “DIK” wordt weergegeven op de led lichten met voorafgaand en achteraf een leuk muziekje wordt gespeeld. Hierbij hebben we met de display.show functie onze eigen image weergegeven die we hebben gemaakt met de getallen 0 en 9 en hebben we deze image voor een seconde in beeld gelaten door de sleep functie. Ook hier waren geen problemen.

### Taakverdeling

Omdat we allemaal geen verstand hadden van robotica zijn we maar met zijn allen aan de slag gegaan. Als de één de code aan het schrijven was, was de ander youtube filmpjes aan het kijken over het onderwerp. Over een taakverdeling valt dus niet echt te spreken. We hebben het verslag ook met zijn allen geschreven. Hier is een globaal overzicht van wie wat heeft gedaan:

bouwen robot	Anne, Mette
code bedenken	Anne, Mette, Wies, Rana en Liv
code schrijven	Mette, Wies, Rana en Liv
informatie opzoeken	Anne, Mette, Wies, Rana en Liv
robot testen	Mette, Wies, Rana en Liv
verslag schrijven	Anne, Mette, Wies, Rana en Liv