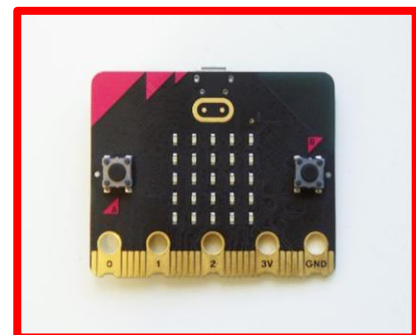




# Mocktailmixer met de TI-Nspire en BBC micro:bit

Leerkrachtenbundel

*Evelyn Blocken,  
Ann-Kathrin Coenen  
& Natalie Dirckx*

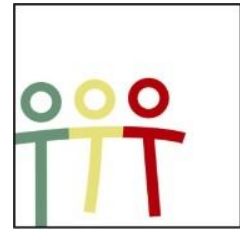


# Inhoudsopgave

<b>Inhoudsopgave.....</b>	<b>2</b>
<b>T<sup>3</sup>-Vlaanderen en T<sup>3</sup>-Nederland.....</b>	<b>3</b>
<b>Inleiding .....</b>	<b>3</b>
<b>Introductie van de BBC micro:bit.....</b>	<b>4</b>
<b>Programmeren met de TI-Nspire CX II .....</b>	<b>5</b>
<b>De benodigdheden .....</b>	<b>6</b>
<b>Werkwijze – bouw de mixer .....</b>	<b>8</b>
<i>Aansluiting TI-Nspire op micro:bit.....</i>	<i>9</i>
<b>Werkwijze - programmeren van de BBC micro:bit en TI-Nspire.....</b>	<b>10</b>
<i>Eenvoudige code.....</i>	<i>10</i>
De code.....	10
<i>Complexere code .....</i>	<i>12</i>
Code.....	13

## T<sup>3</sup>-Vlaanderen en T<sup>3</sup>-Nederland

Evelyn Blocken, Ann-Kathrin Coenen en Natalie Dirckx zijn als wetenschapsleerkrachten werkzaam aan het Agnetencollege Peer. Ze behoren ook tot het lerarennetwerk van T<sup>3</sup> Vlaanderen dat nauw samenwerkt met Nederland. T<sup>3</sup> staat voor Teachers Teaching with Technology. Het doel van deze organisatie is om de professionalisering van leerkrachten op het gebied van ICT en technologie in het onderwijs te bevorderen met gebruik van de technologie van Texas Instruments.



**T<sup>3</sup> VLAANDEREN**

Figuur 1:  
[www.t3vlaanderen.be](http://www.t3vlaanderen.be)

Het is zeker de moeite waard om eens te snuisteren op de website, waar talloze kant-en-klare lesonderwerpen terug te vinden zijn.

## Inleiding

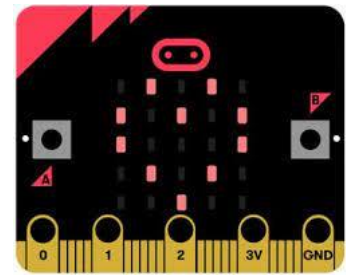
Stel je voor ... een apparaat dat met één druk op de knop de meest verrukkelijke en kleurrijke mocktails voor je mixt. Dit is precies wat deze mocktailmixer doet. Er wordt wel gewerkt met gekleurd water, maar het resultaat is even oogverblindend.

Dit project kan opgedeeld worden in een ontwerpend deel en een programmerend deel. Het bouwen van een mocktailmixer en het programmeren ervan met Python op de TI-Nspire past ideaal binnen de lessen STEM of techniek. De leerlingen leren hierbij het concept van de te mixen kleuren in kleine onderdelen opbreken om de code te kunnen schrijven. En het kan aangewend worden om de begrippen mechanische systemen en elektronica toe te passen op dit project.

## Introductie van de BBC micro:bit

De BBC micro:bit is een populair, op zakformaat computertje of microcontroller. Het is een interface voor de samenwerking tussen software en hardware.

De micro:bit heeft een 5X5 LED lichtdisplay, drukknoppen A en B, aanraak invoerknoppen, ingebouwde microfoon en luidspreker. Daarnaast bevat deze microcontroller zelf nog vele sensoren onder andere voor temperatuur, licht, beweging en een kompas. Tot slot is interactie met andere toestellen of het internet ook mogelijk door middel van een bluetooth verbinding.



Figuur 2: BBC micro:bit

Ben je zelf niet bekend met de onderdelen van de micro:bit? Voor de leerlingen hebben we een BookWidget uitgewerkt. Via deze [Lerarenlink](#) kan je deze widget zelf gebruiken.

De micro:bit voert acties uit na het programmeren van instructies. Deze instructies worden in de programmeertaal Python geschreven op het grafische rekenmachine TI-84 Plus CE-T Python Edition of TI-Nspire CX II (Figuur 4). In deze bundel werd gekozen voor de Nspire om dit project uit te voeren.



Figuur 3: python programmeertaal

Python is een open source programmeertaal die eenvoudig en duidelijk is en toch breed toepasbaar in technologieën. Voor beginners wordt programmeren in python aangeraden en dat maakt deze programmeertaal zeer geschikt voor leerlingen.

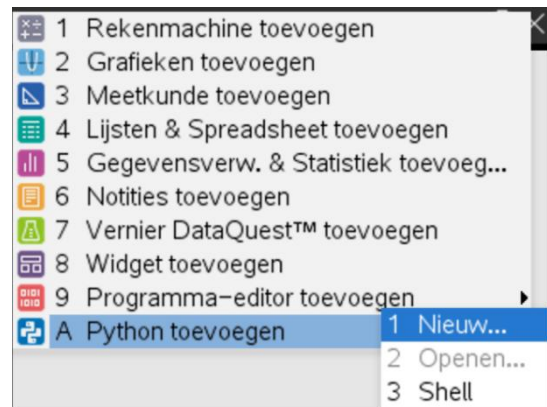


Figuur 4: TI-Nspire CX II

## Programmeren met de TI-Nspire CX II

De TI-Nspire CX II is een grafische rekenmachine met praktische leermiddelen voor zowel de wiskunde- als de natuurwetenschappenlessen. Dit is zowel via de software als handheld te gebruiken.

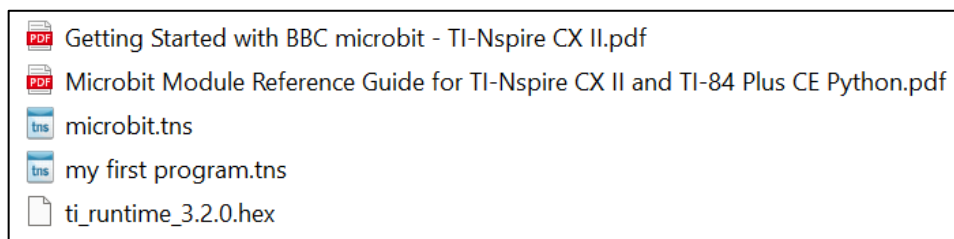
Voor dit project zal er gebruikgemaakt worden van de python module. De code kan zowel op de laptop als de handheld geprogrammeerd worden. Een nieuwe pythoncode schrijf je door in het home-menu een nieuw document te maken en vervolgens 'Python toevoegen' te selecteren (Figuur 5). Met behulp van de menu-knop is het mogelijk om deels voorgeschreven stukken code toe te voegen.



Figuur 5: een nieuw python-bestand maken op de TI-Nspire

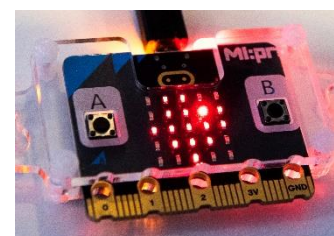
Zowel de micro:bit als de handheld moeten van een module voorzien worden vooraleer het mogelijk is om te kunnen programmeren voor de micro:bit.

Via de website van [TI education](https://www.ti.com/education) kan je de nodige files (Figuur 6) downloaden als zip. Alle nodige bestanden vind je in deze map inclusief een stappenplan voor installatie. Na installatie van microbit.tns is het mogelijk om bepaalde functies van de micro:bit te kunnen gebruiken in de Python pagina van de handheld.



Figuur 6: microbit.tns bevat de module voor de handheld, ti\_runtime.hex is de module voor de microbit

Op de micro:bit dient een hex-file geïnstalleerd te worden. Wanneer de code succesvol op de micro:bit wordt geplaatst, zal het Texas Instruments logo verschijnen. De micro:bit kan via de USB mini naar micro kabel verbonden worden met de TI-Nspire.

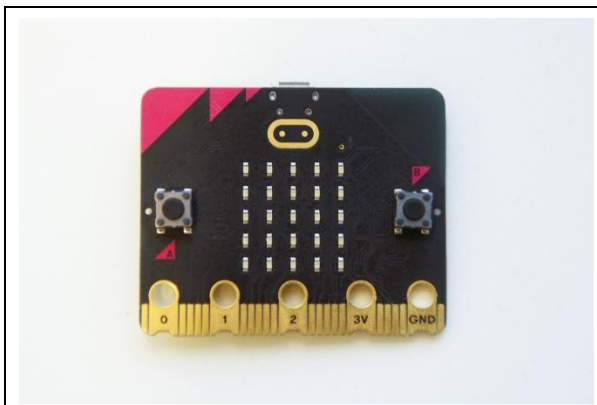


Figuur 7: Texas Instruments logo op display micro:bit

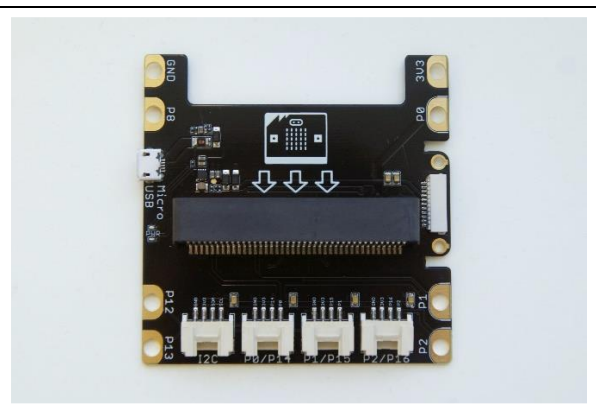
## De benodigheden

Voor één moctailmmixer zijn de volgende benodigheden vereist:

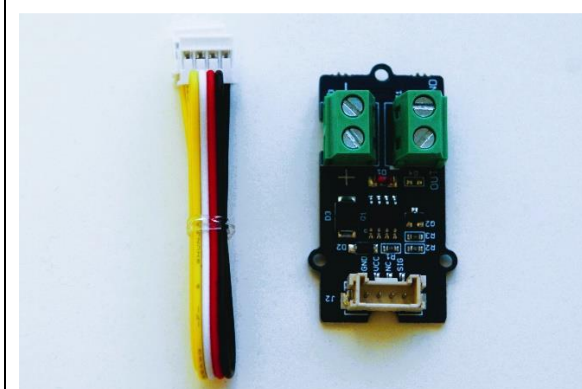
- TI-Nspire CX II
- Micro:bit v2
- Mini-USB naar micro-USB kabel
- Grove shield voor micro:bit v2.0
- Grove Mosfet
- 3x Seeed grove 4 pin kabel
- 3x Batterijhouders 4x AA batterijhouder met aan/uit schakelaar + nodige batterijen
- 3x Kitronik 3V Vertical Submersible Water pomp
- Plastieken buisje voor pompje
- Wijnkist voor 3 flessen
- 3x glazen bokalen
- Gele, rode en blauwe kleurstof



BBC micro:bit



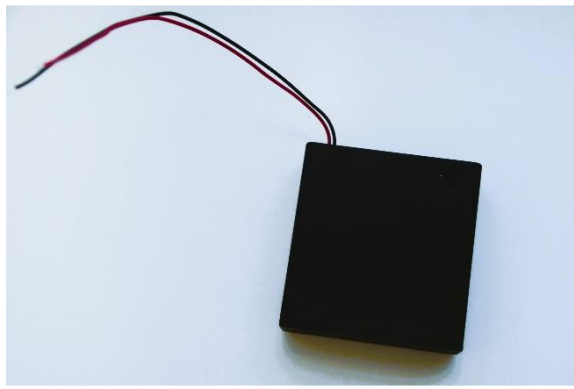
Grove shield voor micro:bit v2.0



Grove Mosfet



Kitronik 3V Vertical Submersible Water Pump



Batterijhouders 4x AA batterijhouder met aan/uit schakelaar

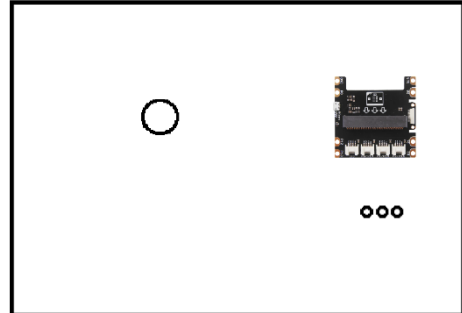


Wijnkist voor 3 flessen

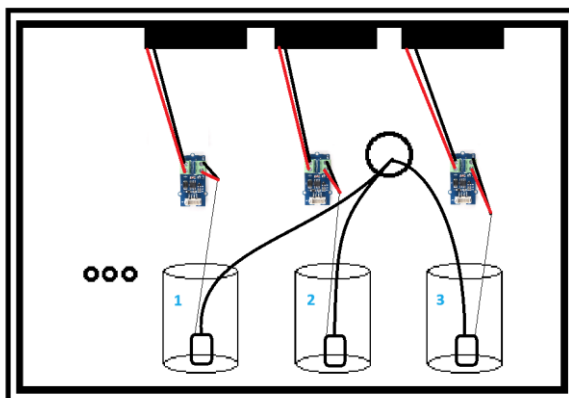
## Werkwijze – bouw de mixer

Het bouwen van de mocktailmixer gebeurt als volgt:

1. Neem de wijnkist met de dichte zijde naar voor en boor 3 gaatjes van een diameter van 1cm langs elkaar aan de rechter voorzijde op ongeveer 10 cm hoogte (Figuur 8).
2. Boor nu een groter gat met een diameter van 3 cm aan de linker voorzijde op ongeveer 20 cm hoogte (Figuur 8).
3. Neem de batterijhouders en plaats de batterijen hierin. Sluit de rode draad van de batterijhouder op de '+' poort aan van de mosfet en de zwarte draad op de '-' poort.
4. Neem het waterpompje en sluit de rode draad op de 'out' poort aan van de mosfet en de zwarte draad op de 'GND' poort. Doe dit voor alle drie de pompjes, de batterijhouders en de mosfets.
5. Draai de wijnkist om en verwijder het schuifdeurtje. Bevestig de batterijhouders met dubbelzijdige tape aan de onderzijde van de bovenkant van de kist.
6. Bevestig de 3 mosfets aan de achterzijde van de kist boven de glazen bokalen.
7. Zet de 3 glazen bokalen in de kist en plaats steeds één waterpompje in elke bokaal.
8. Knip 3 plasticen buisjes. Plaats op elk pompje een buisje en laat deze samen aan de voorkant doorheen het grotere gat van 3 cm uitkomen (Figuur 9).
9. Bevestig de grove shield aan de voorzijde van de kist boven de drie gaatjes (Figuur 8).
10. Neem de grove 4 pin kabels en steek steeds één door elk gaatje.
11. Bevestig steeds één grove 4 pin kabel op één mosfet.
12. Bevestig de kabel van het pompje in bokaal 1 op pin 0 van de grove shield, de kabel van het pompje in bokaal 2 op pin 1 en de kabel van het pompje in bokaal 3 op pin 2.
13. Vul alle bokalen met water. Voeg aan bokaal 1 gele kleurstof toe, aan bokaal 2 rode kleurstof en aan bokaal 3 blauwe kleurstof (Figuur 9).



Figuur 8: het maken van de gaten in de wijnkist en de plaatsing van de grove shield



Figuur 9: aansluiting van de pompjes, mosfets en batterijhouders



## Aansluiting TI-Nspire op micro:bit

Wanneer de mocktailmixer is afgewerkt, kan de micro:bit aangesloten worden met de usb-kabel mini naar micro volgens Figuur 11. In Figuur 10 staat een volledige opstelling afgebeeld.



Figuur 11: mini naar micro usb-kabel



Figuur 10: een afgewerkte mocktailmixer

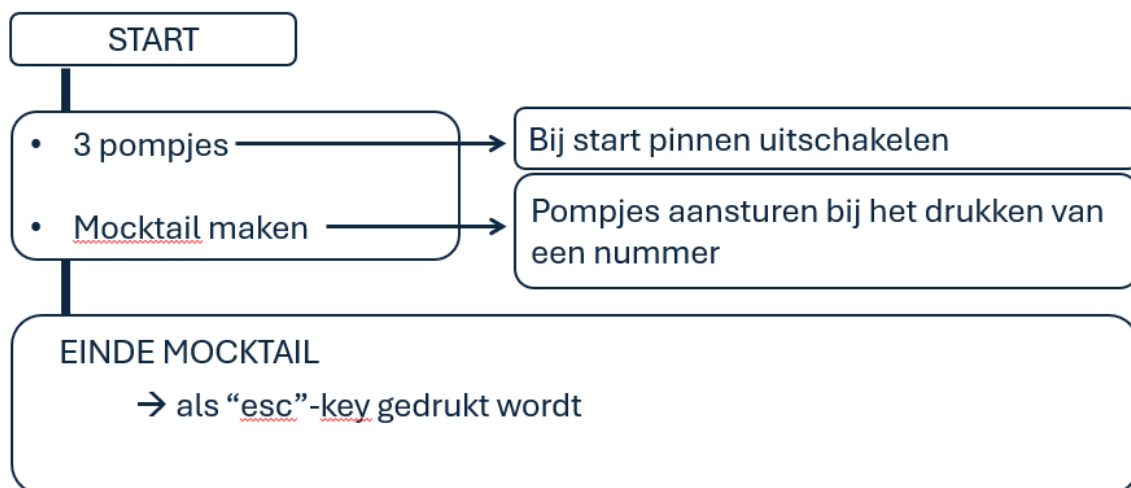
## Werkwijze - programmeren van de BBC micro:bit en TI-Nspire

Om leerlingen die nog nooit geprogrammeerd hebben te begeleiden in het leerproces, breken we de code in kleine stappen op.

Afhankelijk van de moeilijkheidsgraad kan je de mocktailmixer op 2 verschillende manieren programmeren, namelijk een eenvoudige code en een complexere code. De verschillende mogelijkheden worden hieronder verder toegelicht.

### Eenvoudige code

Voor startende leerlingen kan een eenvoudige code opgebouwd worden waarin je 3 mocktails kan maken. Het doel van deze code is de 3 pompjes aansturen bij een bepaalde input en na een bepaald verloop van tijd weer uitschakelen. Aan ieder pompje kan een gekleurde vloeistof en dus kleur toegekend worden, bijvoorbeeld rood, blauw en geel. Een voorbeeld van dit codeerschema wordt hieronder weergegeven (Figuur 12).



Figuur 12: Codeerschema eenvoudige code

### De code

```
from microbit import *
```

Allereerst maakt 'from microbit import\*' het mogelijk om de library code voor micro:bit te gebruiken.

```
pin0.write_digital(0)  
pin1.write_digital(0)  
pin2.write_digital(0)
```

De pompjes mogen enkel aangaan wanneer dit gevraagd wordt. Door de pin waarop ze zijn aangesloten in stand 0 te zetten, worden ze aan het begin op inactief gezet.

```
print ("Welke mocktail wilt u maken? Voer hier het nummer in")
while get_key() != "esc":
    x=get_key()
```

Wanneer de mocktailmixer opgestart is zal op het display van de Nspire de boodschap “Welke mocktail wilt u maken? Voer hier het nummer in” verschijnen door middel van de print-functie.

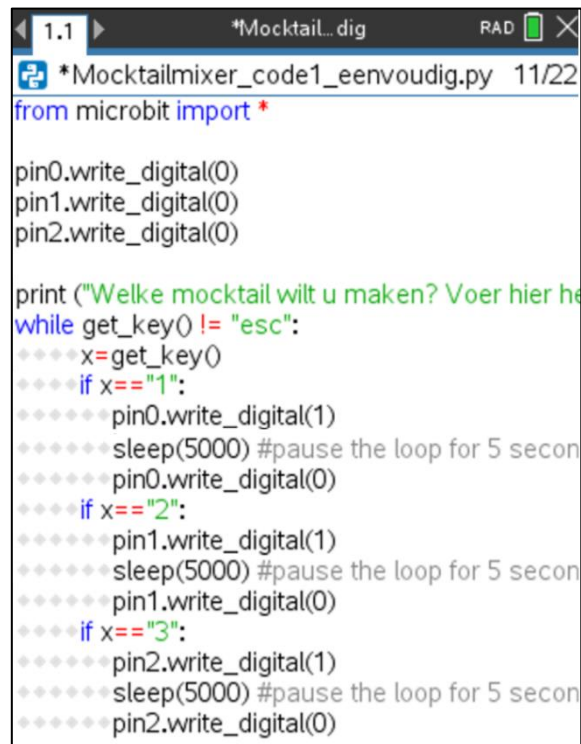
Het doel van de mocktailmixer is constant mocktails kunnen maken zolang de code loopt. Dit betekent dat we ook een “afsluit”-knop wil toevoegen, namelijk “esc”. Met de code ‘while get\_key() != “esc” ‘ zal de while-lus blijven loop totdat de esc-knop op de Nspire ingedrukt wordt.

Het getKey commando geeft de waarde van de laatst ingedrukte toets sinds de laatste keer dat getKey werd uitgevoerd. De waarde van getKey wordt gewist elke keer dat je ervan leest, totdat een nieuwe toets wordt ingedrukt. Om deze reden wordt de waarde van getKey eerst opgeslagen in een variabele, in dit geval “x”.

```
if x=="1":
    pin0.write_digital(1)
    sleep(5000) #pause the loop for 5 seconds
    pin0.write_digital(0)
if x=="2":
    pin1.write_digital(1)
    sleep(5000) #pause the loop for 5 seconds
    pin1.write_digital(0)
if x=="3":
    pin2.write_digital(1)
    sleep(5000) #pause the loop for 5 seconds
    pin2.write_digital(0)
```

Wanneer een bepaalde toets wordt ingedrukt, zal één pompje moeten aangaan. Zo zal bij toets 1 het pompje op pin 0 aangaan, bij toets 2 het pompje op pin 1 en bij toets 3 het pompje op pin 2. Deze pompjes blijven dan voor 5 seconden aan en pompen de vloeistof naar het glas. Na deze 5 seconden zal het pompje weer uitschakelen. Hierna is de “if”-functie volledig uitgevoerd en kan men een nieuwe toets indrukken en dus een volgende mocktail maken.

In Figuur 13 kan je de code in zijn geheel zien.



```
1.1 *Mocktail... dig RAD
*Mocktailmixer_code1_eenvoudig.py 11/22
from microbit import *

pin0.write_digital(0)
pin1.write_digital(0)
pin2.write_digital(0)

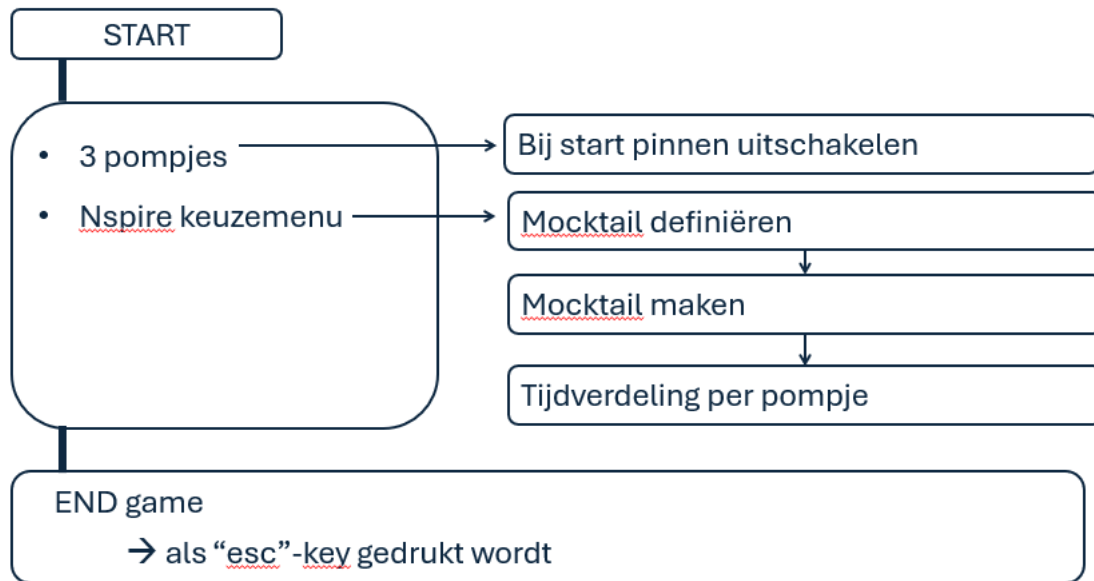
print ("Welke mocktail wilt u maken? Voer hier het nummer in")
while get_key() != "esc":
    x=get_key()
    if x=="1":
        pin0.write_digital(1)
        sleep(5000) #pause the loop for 5 seconds
        pin0.write_digital(0)
    if x=="2":
        pin1.write_digital(1)
        sleep(5000) #pause the loop for 5 seconds
        pin1.write_digital(0)
    if x=="3":
        pin2.write_digital(1)
        sleep(5000) #pause the loop for 5 seconds
        pin2.write_digital(0)
```

Figuur 13: volledige eenvoudige code

Als de code geschreven is, kan deze gerund worden. Dit doe je door in 'menu', 'uitvoeren' te kiezen. Dit kan ook door 'Ctrl'+ 'R' te drukken. In een nieuwe pagina zal een Shell openen. Met 'Ctrl' + pijltje naar links of rechts kan je tussen de pagina's wisselen. Wil je iets in je code veranderen, zal je dus op de vorige pagina je aanpassing moeten maken en het programma opnieuw runnen in de shell.

### Complexere code

In een complexere code is het ook mogelijk om verschillende kleuren te laten mengen en zo tot wel 7 mocktails te laten mixen. Hierin moeten verschillende pompjes om de beurt in werking treden na het geven van een input. Daarnaast wordt deze code ook nog uitgebreid met een boodschap op de Nspire. Het display van deze wordt aangestuurd om de mogelijke mocktails weer te geven bij de start en een melding weer te geven wanneer deze gemaakt worden. Een voorbeeld van dit codeerschema wordt hieronder weergegeven (Figuur 14).



Figuur 14: Codeerschema complexe code

## Code

```
from microbit import *
import ti_plotlib as plt
```

Allereerst maakt 'from microbit import\*' het mogelijk om de library code voor micro:bit te gebruiken. Met 'import ti\_plotlib as plt' is het mogelijk om het display van de Nspire te programmeren als scherm met informatie/vragen.

```
pin0.write_digital(0)
pin1.write_digital(0)
pin2.write_digital(0)
```

De pompjes mogen enkel aangaan wanneer dit gevraagd wordt. Door de pin waarop ze zijn aangesloten in stand 0 te zetten, worden ze aan het begin op inactief gezet.

```
x=get_key()
mocktail1 = "Hub on the beach"
mocktail2 = "Bloody instruments"
mocktail3 = "Nspire yourself"
mocktail4 = "Tripple T sunrise"
mocktail5 = "Rover tonic"
mocktail6 = "Texas lover"
mocktail7 = "Wil-de-martini"
```

De waarde van getKey wordt eerst opgeslagen in een variabele namelijk "x". De namen worden aan de 7 verschillende mocktails gekoppeld. De nummers van de mocktail zullen uiteindelijk ook overeenkomen met de toets die hiervoor ingedrukt moet worden.

```

def startscherm():
    >>>> plt.cls()
    >>>> plt.text_at(2, "Welke mocktail wilt u maken?")
    >>>> plt.text_at(3, "Voer hier het nummer in:", "center")
    >>>> plt.text_at(5, "1. " + mocktail1, "center")
    >>>> plt.text_at(6, "2. " + mocktail2, "center")
    >>>> plt.text_at(7, "3. " + mocktail3, "center")
    >>>> plt.text_at(8, "4. " + mocktail4, "center")
    >>>> plt.text_at(9, "5. " + mocktail5, "center")
    >>>> plt.text_at(10, "6. " + mocktail6, "center")
    >>>> plt.text_at(11, "7. " + mocktail7, "center")

```

Om te veel herhaling in de code te vermijden, wordt de programmatie van het startscherm in een functie geplaatst. De functie wordt gedefinieerd als startscherm. Dit scherm zal steeds na het opstarten van de code of het maken van een cocktail op de Nspire getoond worden. Hierbij verschijnt steeds de boodschap "Welke cocktail wilt u maken?", hieronder "Voer hier het nummer in:" en tot slot een lijst met alle mogelijke mocktails en het bijhorende nummer.

```

def mocktail(naam, pin0ms, pin1ms, pin2ms):
    >>>> plt.cls()
    >>>> plt.text_at(2, "Uw " + naam + " wordt bereid", "center")
    >>>> if pin0ms != 0:
    >>>>>> pin0.write_digital(1)
    >>>>>> sleep(pin0ms)
    >>>>>> pin0.write_digital(0)
    >>>> if pin1ms != 0:
    >>>>>> pin1.write_digital(1)
    >>>>>> sleep(pin1ms)
    >>>>>> pin1.write_digital(0)
    >>>> if pin2ms != 0:
    >>>>>> pin2.write_digital(1)
    >>>>>> sleep(pin2ms)
    >>>>>> pin2.write_digital(0)
    >>>> plt.text_at(4, "Uw " + naam + " is klaar!", "center")
    >>>> plt.text_at(5, "Smakelijk!", "center")
    >>>> sleep(5000)
    >>>> startscherm()

```

Ook het maken van de mocktail wordt in een functie opgebouwd om te veel herhaling te vermijden. Deze wordt gedefinieerd als mocktail waarbij verschillende argumenten kunnen veranderen, zoals de naam van de mocktail en de tijd dat de 3 verschillende pompjes al dan niet ingeschakeld worden.

Wanneer deze functie wordt uitgevoerd, zal het startscherm van de Nspire leeggemaakt worden en de boodschap "Uw ##naam van de mocktail## wordt gemaakt verschijnen". Indien een pompje in het argument van de functie een activatietijd heeft gekregen, zal deze waarde dus niet gelijk zijn aan 0 en zal dit pompje voor de tijd aanschakelen. Nadat deze tijd verlopen is, zal deze ook weer uitschakelen.

Nadat de mocktail gemaakt is, zal het display van de Nspire de boodschap “Uw *##naam van mocktail##* is klaar” en eronder “Smakelijk!”. Deze boodschap zal voor 5 seconden blijven staan waarna het startscherm weer zal verschijnen.

```
startscherm()
while x != "esc":
    x=get_key()
    if x=="1":
        mocktail(mocktail1,5000,0,0)
    if x=="2":
        mocktail(mocktail2,0,5000,0)
    if x=="3":
        mocktail(mocktail3,0,0,5000)
    if x=="4":
        mocktail(mocktail4,2500,2500,0)
    if x=="5":
        mocktail(mocktail5,2500,0,2500)
    if x=="6":
        mocktail(mocktail6,0,2500,2500)
    if x=="7":
        mocktail(mocktail7,1600,1600,1600)
```

Nadat het startscherm wordt weergegeven, wordt er een mocktail gekozen. Zolang de esc-knop niet wordt ingedrukt, kunnen er mocktails gemaakt worden. De waarde van getKey wordt eerst opgeslagen in een variabele namelijk “x”. Afhankelijk welke knop ingedrukt wordt, zal de functie mocktail uitgevoerd worden. Hiervoor wordt gedefinieerd welke mocktail er gemaakt wordt en welke pompje voor hoeveel milliseconden moet inschakelen.

In Figuur 15 kan je de code in zijn geheel zien.

```

1.1 1.2 *Mocktail...e 2 RAD 11/66
Microbit_glas_functie.py
from microbit import *
import ti_plotlib as plt

pin0.write_digital(0)
pin1.write_digital(0)
pin2.write_digital(0)

x=get_key()
mocktail1 = "Hub on the beach"
mocktail2 = "Bloody instruments"
mocktail3 = "Nspire yourself"
mocktail4 = "Tripple T sunrise"
mocktail5 = "Rover tonic"
mocktail6 = "Texas lover"
mocktail7 = "Wil-de-martini"

def startscherm():
    plt.cls()
    plt.text_at(2,"Welke mocktail wilt u maken?")
    plt.text_at(3,"Voer hier het nummer in:","center")
    plt.text_at(5,"1. " + mocktail1,"center")
    plt.text_at(6,"2. " + mocktail2,"center")
    plt.text_at(7,"3. " + mocktail3,"center")
    plt.text_at(8,"4. " + mocktail4,"center")
    plt.text_at(9,"5. " + mocktail5,"center")
    plt.text_at(10,"6. " + mocktail6,"center")
    plt.text_at(11,"7. " + mocktail7,"center")

def mocktail(naam,pin0ms,pin1ms,pin2ms):
    plt.cls()
    plt.text_at(2,"Uw "+naam+" wordt bereid","center")
    if pin0ms != 0:
        pin0.write_digital(1)
        sleep(pin0ms)
        pin0.write_digital(0)
    if pin1ms != 0:
        pin1.write_digital(1)
        sleep(pin1ms)
        pin1.write_digital(0)
    if pin2ms != 0:
        pin2.write_digital(1)
        sleep(pin2ms)
        pin2.write_digital(0)
    plt.text_at(4,"Uw " + naam + " is klaar!","center")
    plt.text_at(5,"Smakelijk!","center")
    sleep(5000)
    startscherm()

startscherm()
while x != "esc":
    x=get_key()
    if x=="1":
        mocktail(mocktail1,5000,0,0)
    if x=="2":
        mocktail(mocktail2,0,5000,0)
    if x=="3":
        mocktail(mocktail3,0,0,5000)
    if x=="4":
        mocktail(mocktail4,2500,2500,0)
    if x=="5":
        mocktail(mocktail5,2500,0,2500)
    if x=="6":
        mocktail(mocktail6,0,2500,2500)
    if x=="7":
        mocktail(mocktail7,1600,1600,1600)

```

Figuur 15: volledige uitgebreide code



Als de code geschreven is, kan deze gerund worden. Dit doe je door in 'menu', 'uitvoeren' te kiezen. Dit kan ook door 'Ctrl'+ 'R' te drukken. In een nieuwe pagina zal een Shell openen. Met 'Ctrl' + pijltje naar links of rechts kan je tussen de pagina's wisselen. Wil je iets in je code veranderen, zal je dus op de vorige pagina je aanpassing moeten maken en het programma opnieuw runnen in de shell.