Algorithmen mit dem Calliope Mini

Algorithmen

Ole Vanhoefer

14. September 2025

Entwicklungsumgebung MakeCode

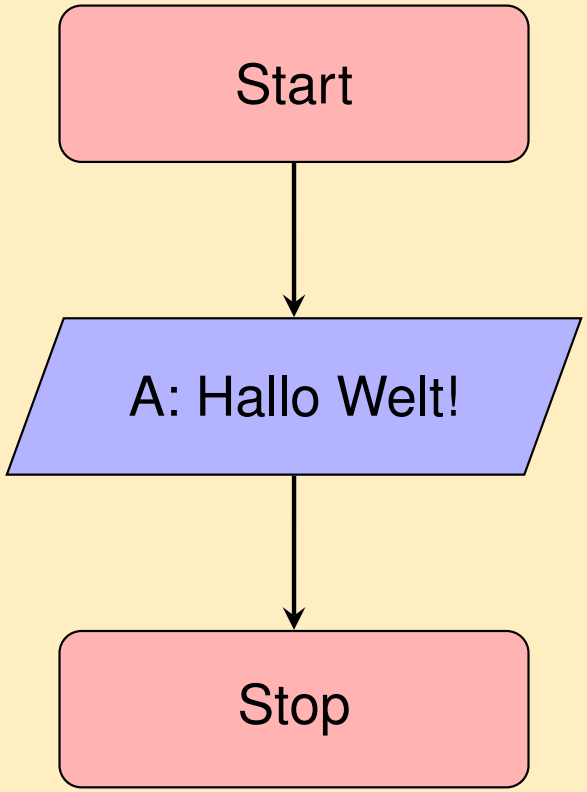
- MakeCode läuft in einem Webbrowser.
- URL: `https://makecode.calliope.cc/`

Anwendung

- Keine Anmeldung nötig.
- **Neues Projekt**: Calliope Typ auswählen
- Speicherung über GitHub möglich.

Ausgabe

Flussdiagramm



Ausgabe

Struktogramm

Ausgabe “Hallo Welt!”

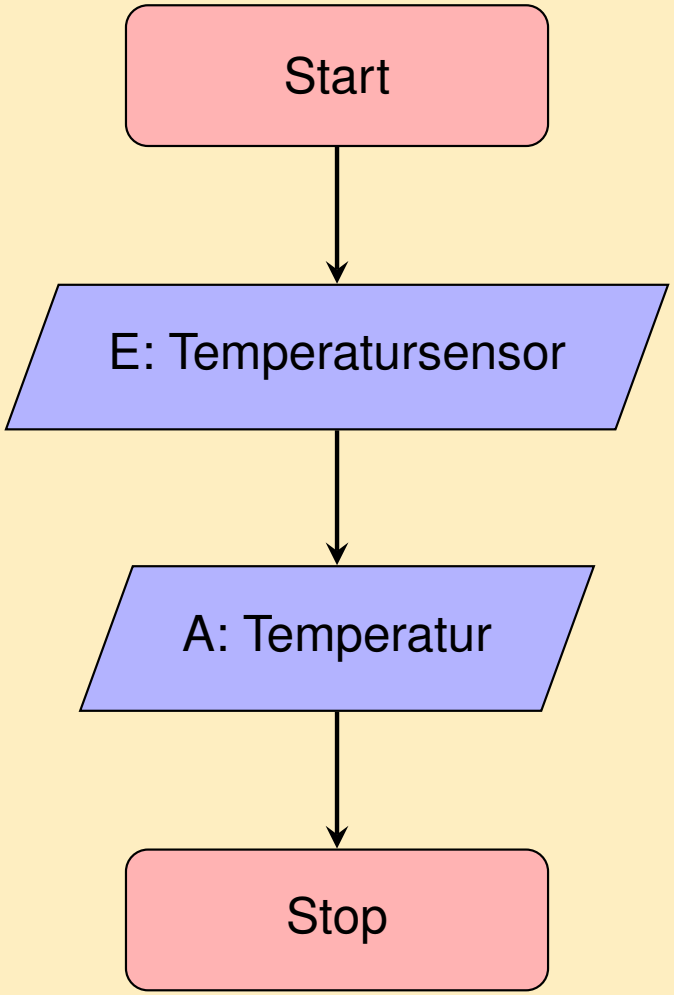
Ausgabe

Pseudocode

- START
- Ausgabe: “Hallo Welt”
- STOP

Eingabe - Ausgabe

Flussdiagramm



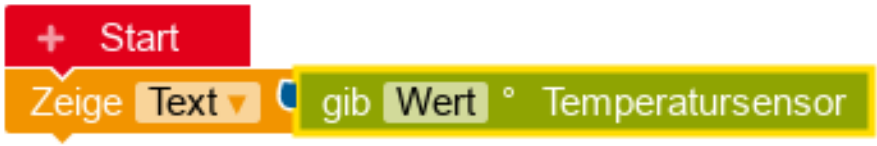
Eingabe - Ausgabe

Struktogramm

Auslesen Temperatursensor
Ausgabe Temperatur

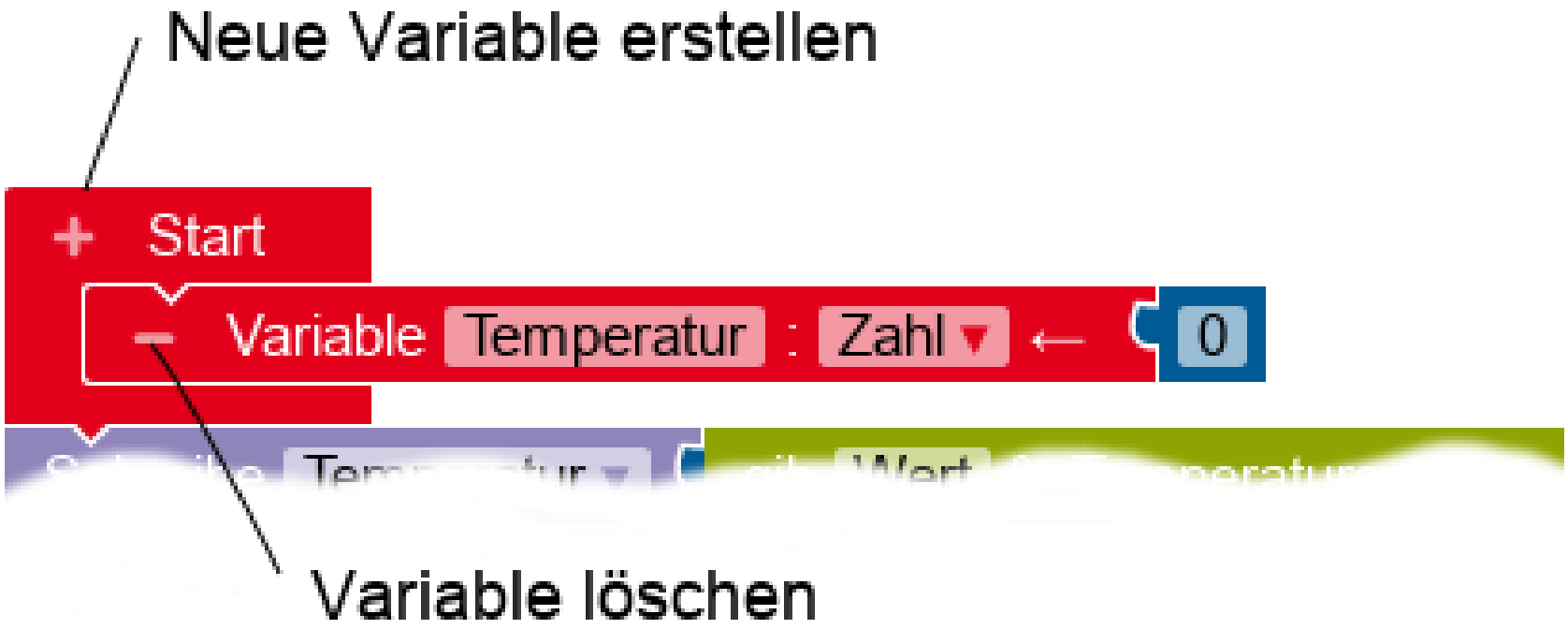
Pseudocode

- START
- Eingabe vom Temperatursensor
- Ausgabe des Temperatursensorwertes
- STOP



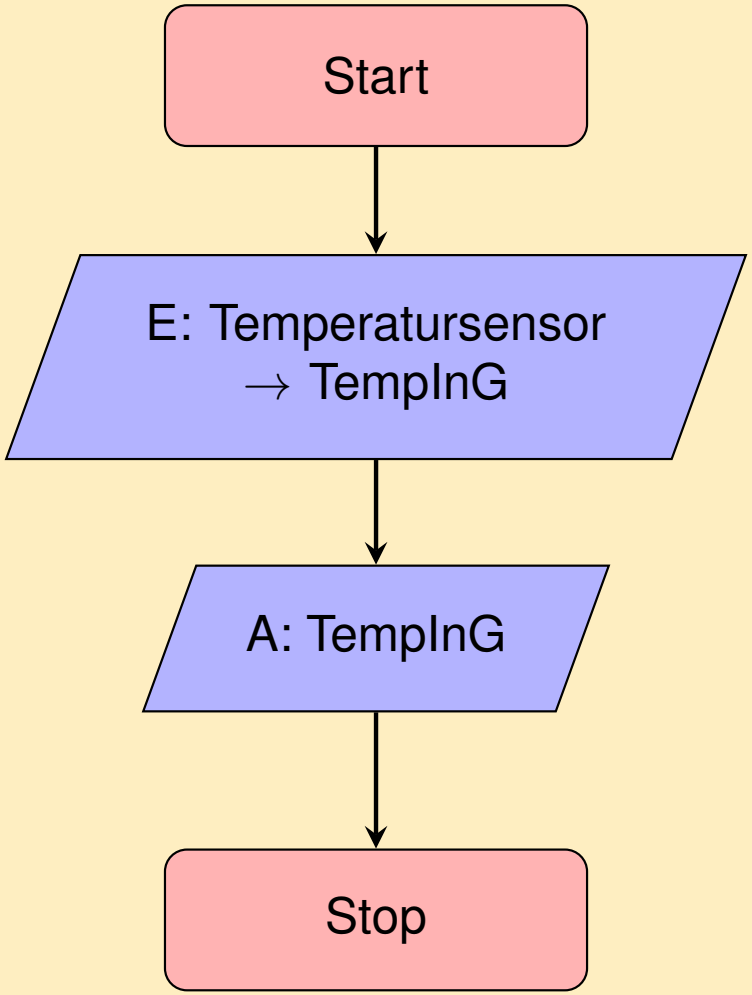
Variablen

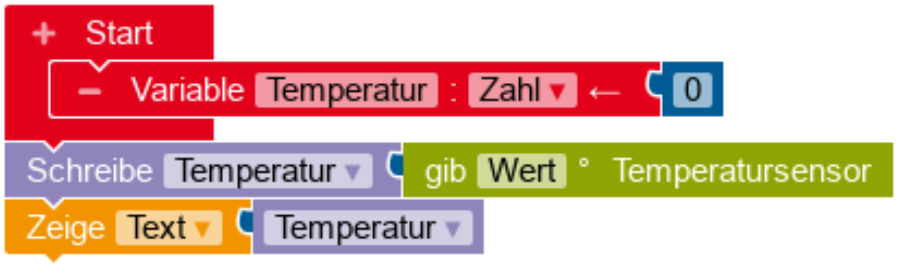
Variablen dienen als Gedächtnis eines Programms. Hier können Werte abgespeichert werden und dann im weiteren Programmverlauf wieder abgerufen werden.



Eingabe - Ausgabe mit Variable

Flussdiagramm

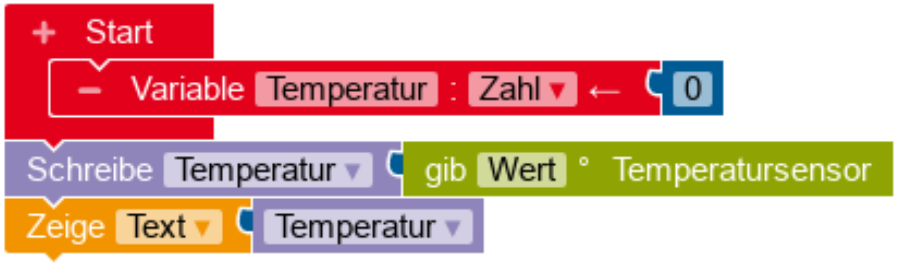




Eingabe - Ausgabe mit Variable

Struktogramm

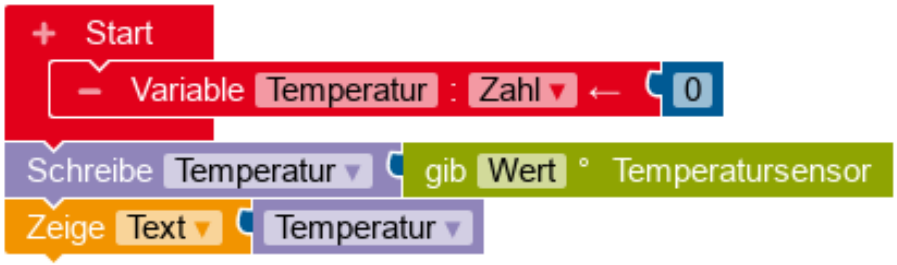
TempInG ← Temperatursensor
Ausgabe TempInG



Eingabe - Ausgabe mit Variable

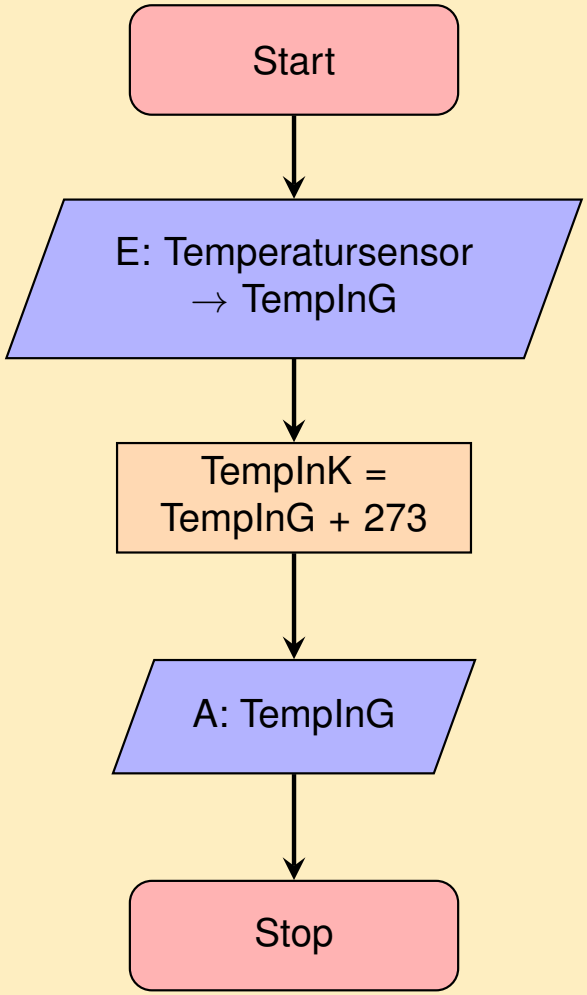
Pseudocode

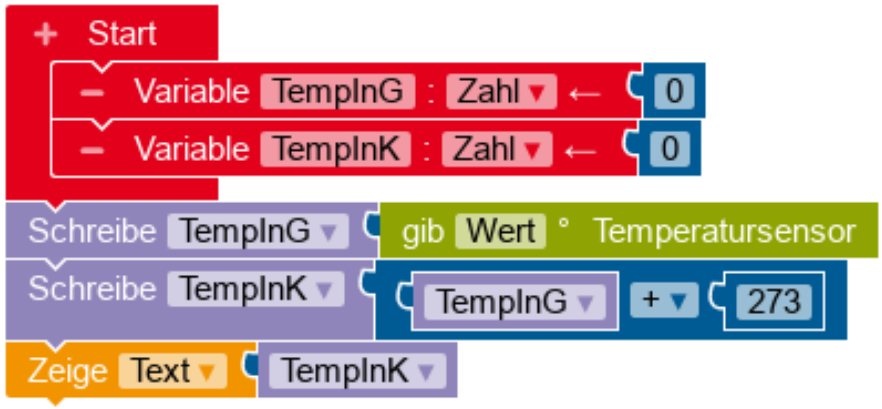
- START
- Variablen
  - Temperatur (Zahl) = 0
- Lese den Wert des Temperatursensors und merke ihn in Temperatur.
- Gebe den Wert der Variablen Temperatur aus.
- STOP



EVA - Eingabe Verarbeitung Ausgabe

Flussdiagramm

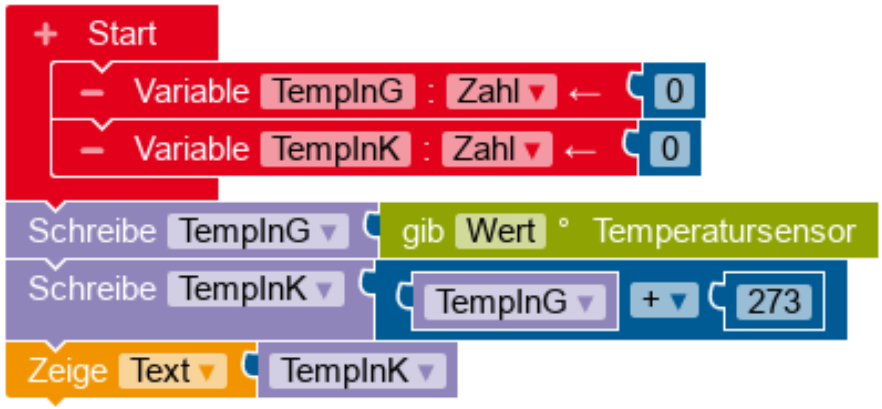




EVA - Eingabe Verarbeitung Ausgabe

Struktogramm

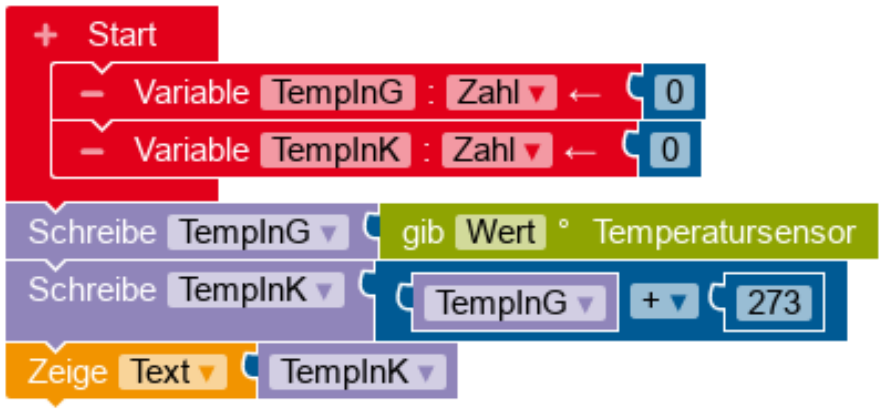
TempInG ← Temperatursensor
TempInK = TempInG + 273
Ausgabe TempInK



EVA - Eingabe Verarbeitung Ausgabe

Pseudocode

- START
- Variablen
  - Temperatur in Grad: TempInG (Zahl) = 0
  - Temperatur in Kelvin: TempInG (Zahl) = 0
- Lese den Wert des Temperatursensors und merke ihn in TempInG.
- TempInK = TempInG + 273
- Gebe den Wert der Variablen TempInK aus.
- STOP



Entscheidungen

Entscheidungen

Wenn die Temperatur über 28 Grad steigt, **dann** gibt es Hitzefrei.

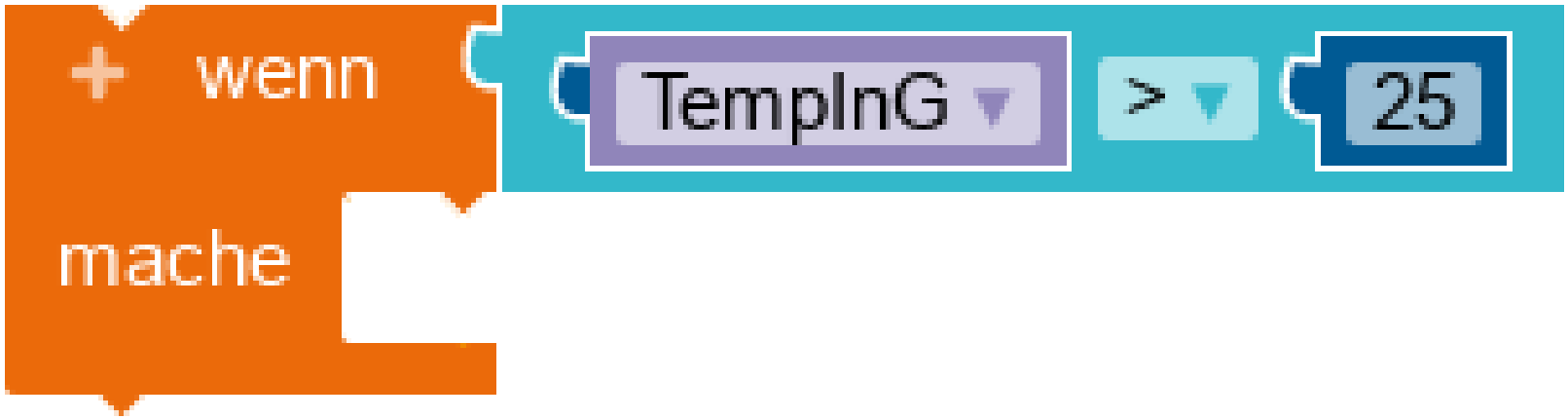
Entscheidungen

Wenn die Temperatur über 28 Grad steigt, **dann** gibt es Hitzefrei.

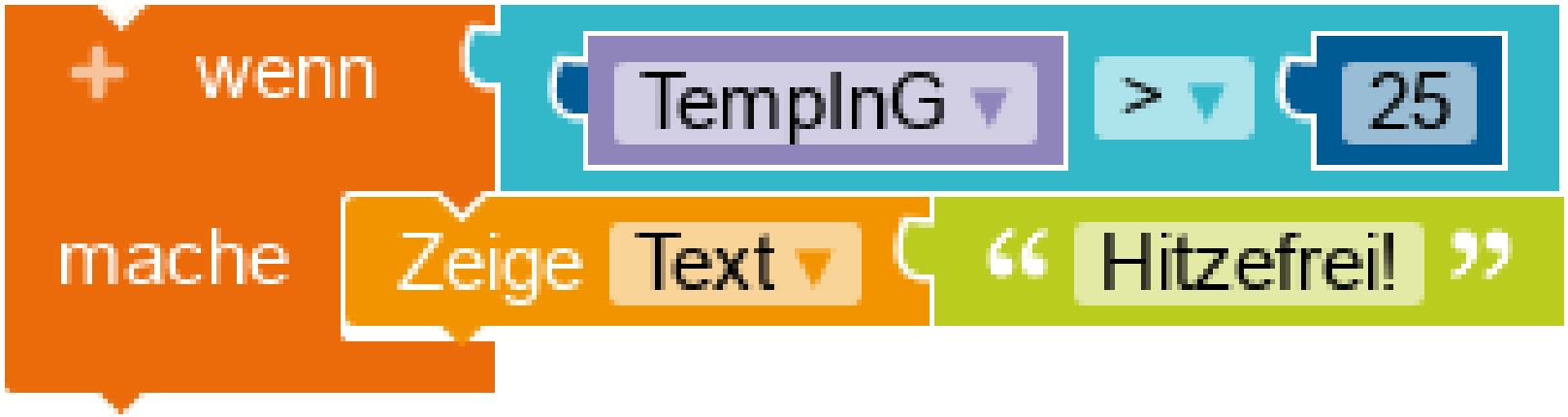
Wenn es morgen regnet, **dann** gehe ich ins Kino.

Entscheidungen

Entscheidungen

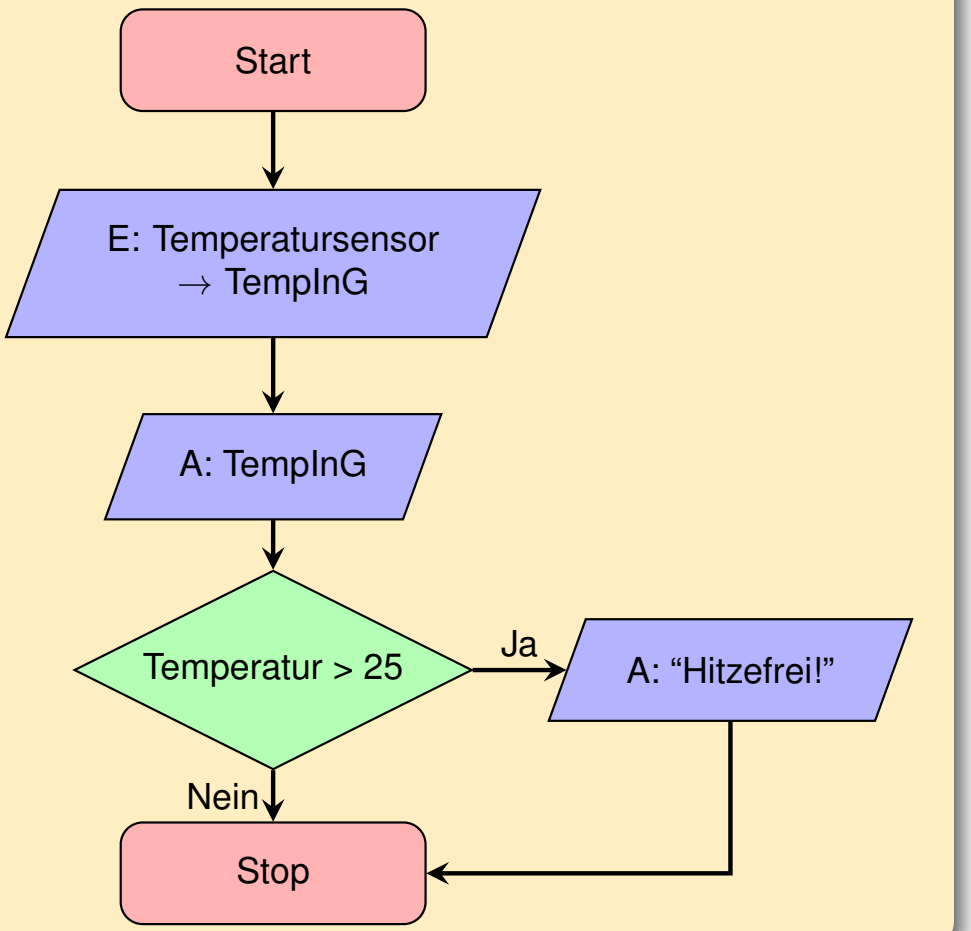


Entscheidungen



EVA - Entscheidung: Wenn ... mache ...

Flussdiagramm



EVA - Entscheidung: Wenn ... mache ...

Struktogramm

TempInG ← Temperatursensor	
Ausgabe TempInK	
Temperatur > 25 (Ja)	Nein
Ausgabe “Hitzefrei”	

Entscheidungen

Entscheidungen

Wenn die Temperatur über 20 Grad steigt, **dann** ziehe das T-Shirt an, **sonst** ziehe den Pullover an.

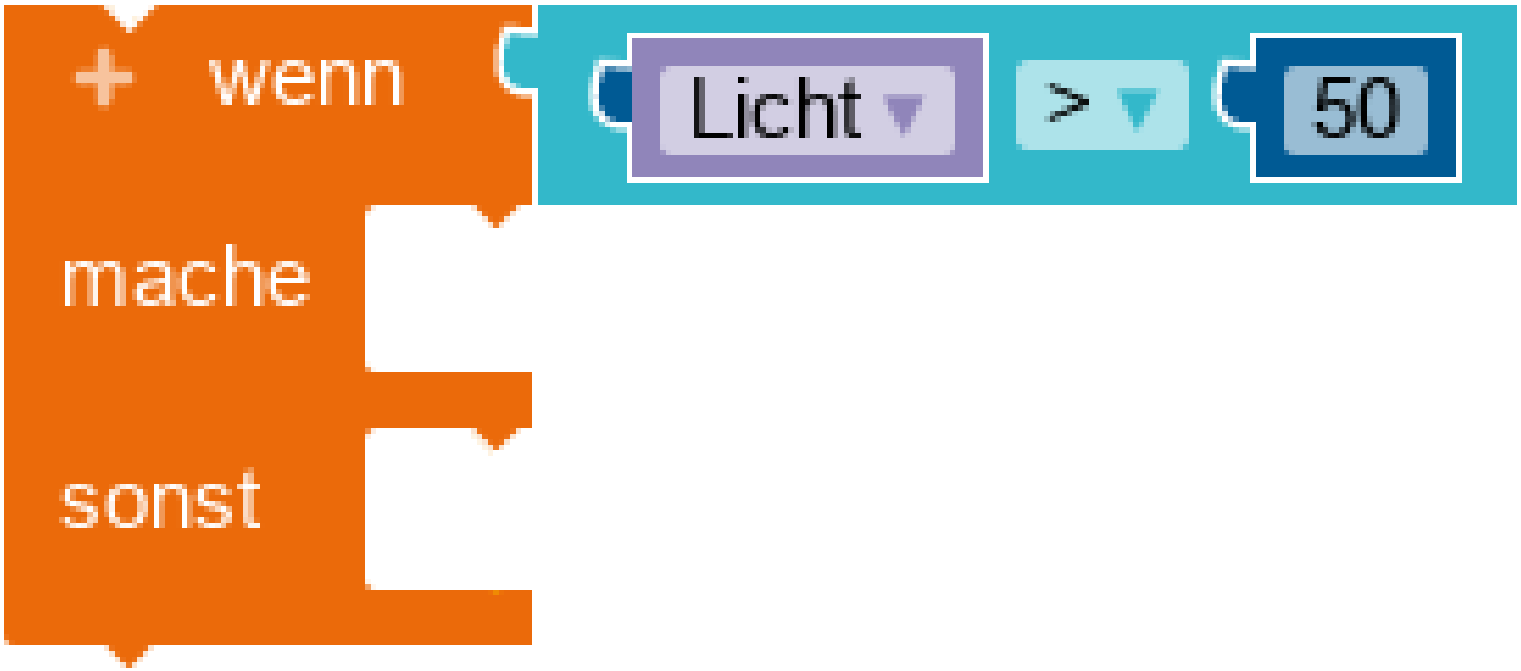
Entscheidungen

Wenn die Temperatur über 20 Grad steigt, **dann** ziehe das T-Shirt an, **sonst** ziehe den Pullover an.

Wenn es morgen regnet, **dann** fahre mit dem Auto, **sonst** fahre mit dem Fahrrad.

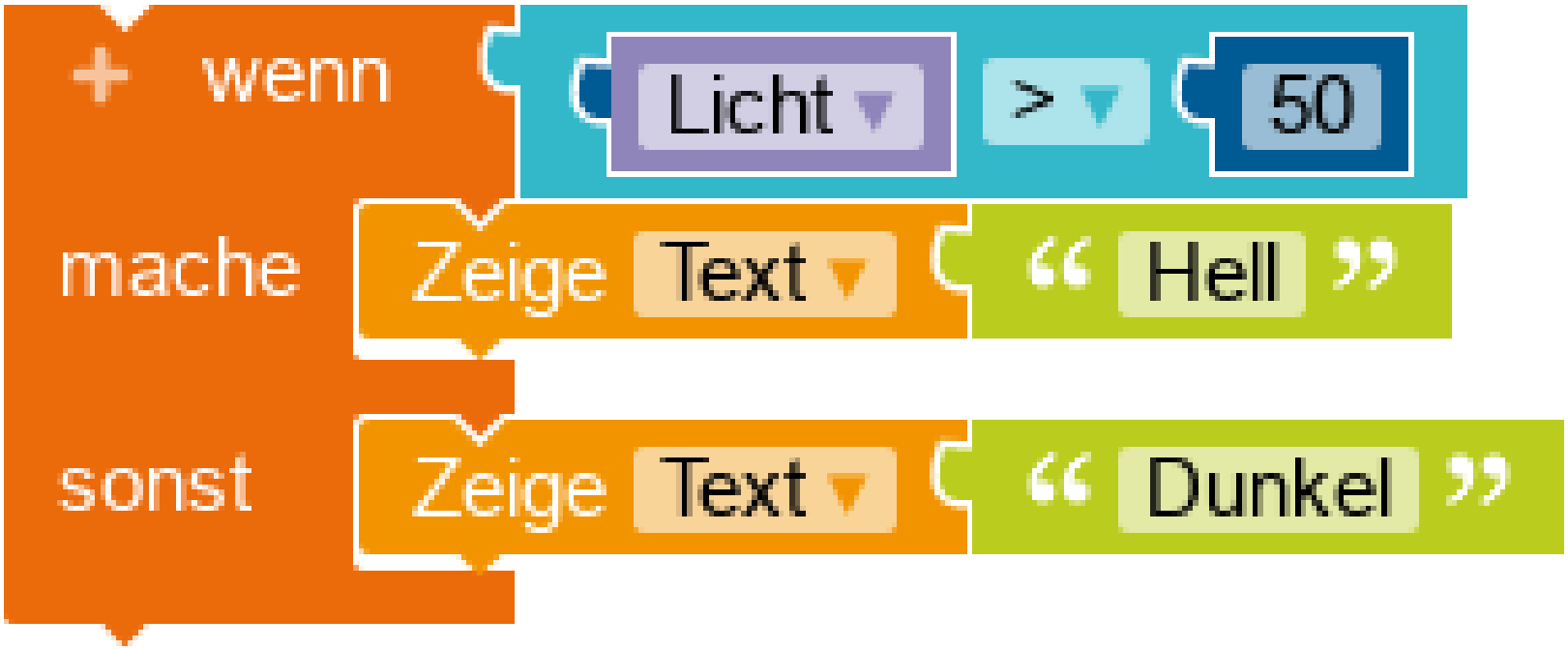
Entscheidungen

Entscheidungen



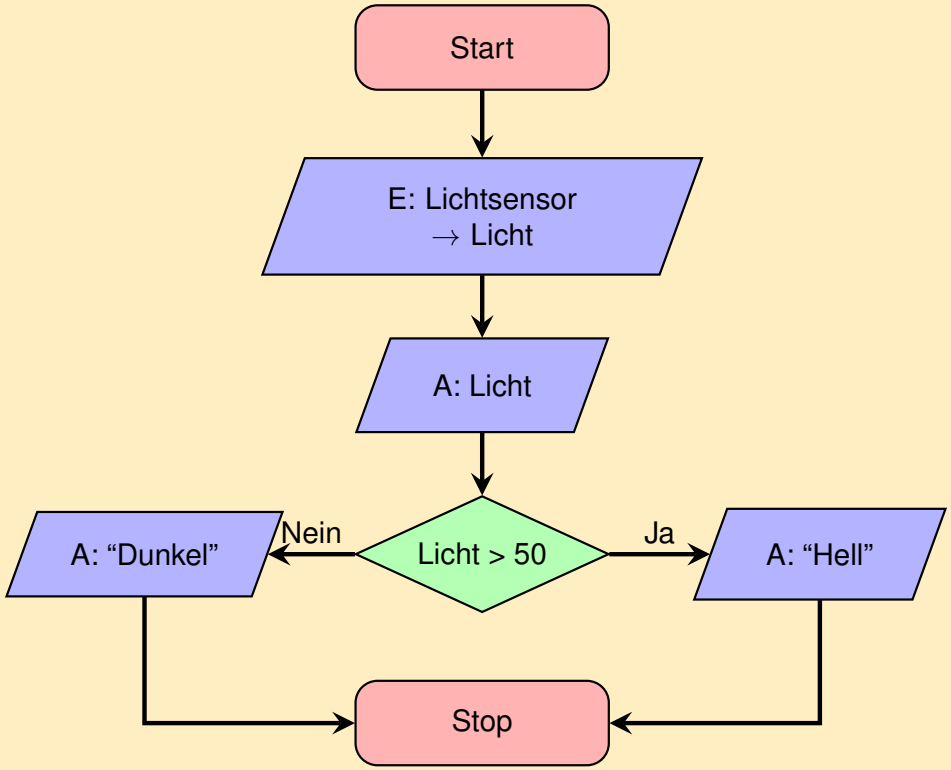
Entscheidungen

Entscheidungen



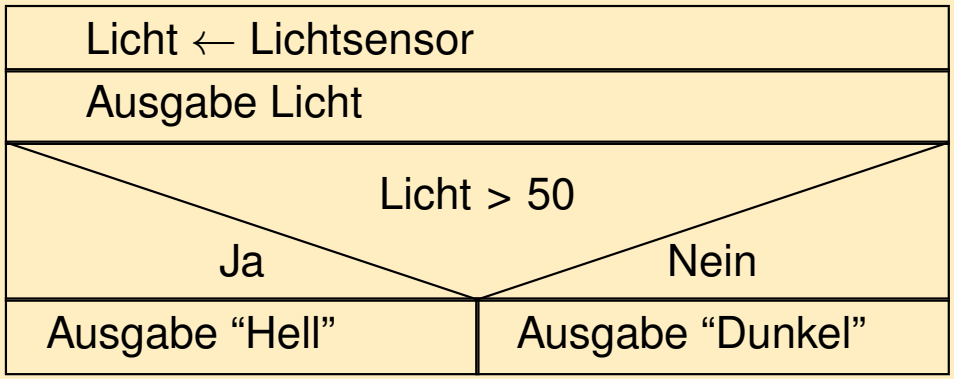
EVA - Wenn ... mache ... sonst ...

Flussdiagramm



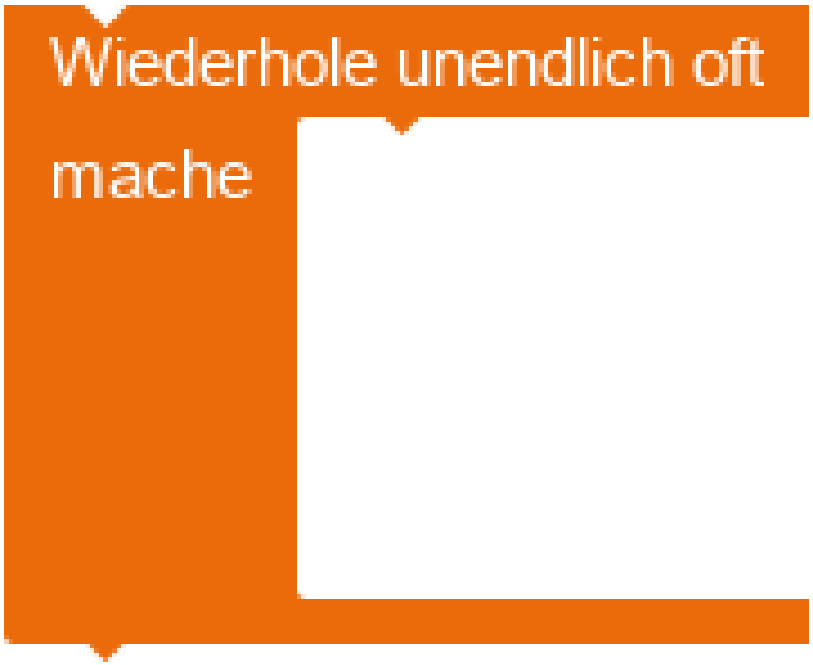
EVA - Wenn ... mache ... sonst ...

Struktogramm

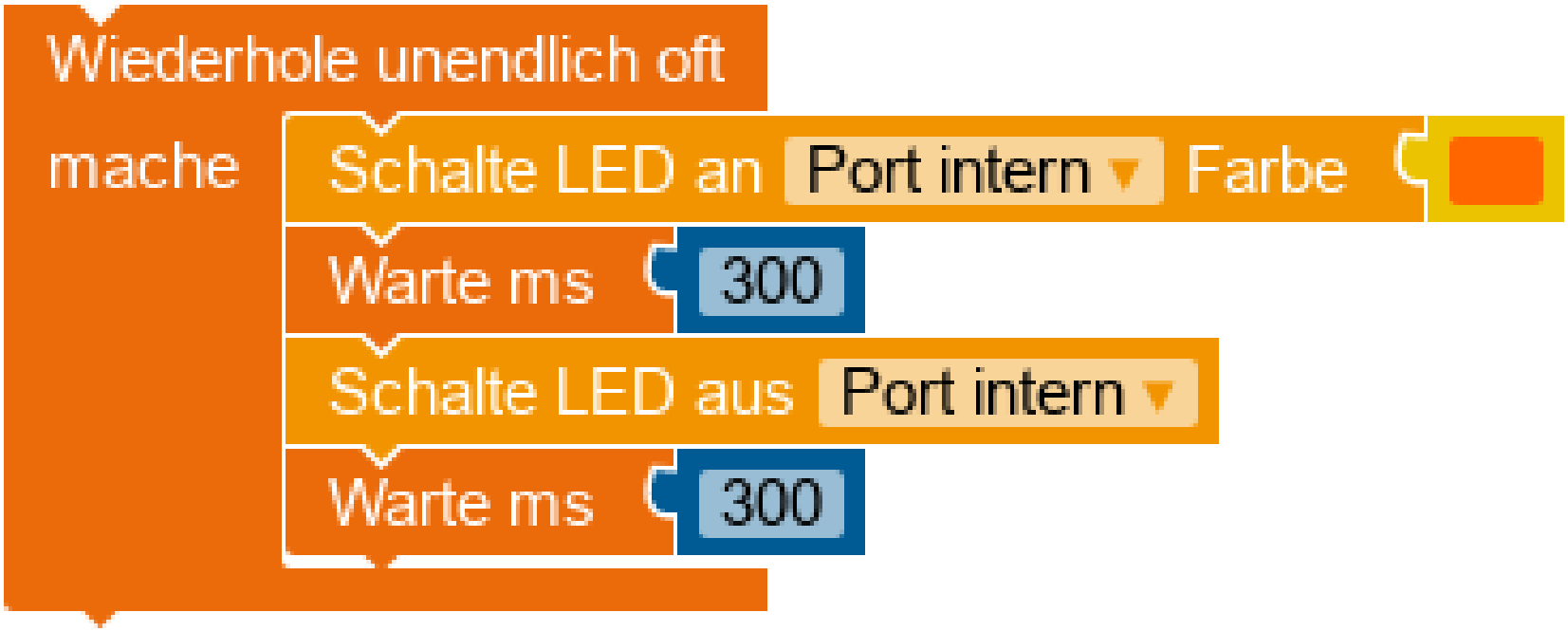


Endlosschleife

Endlosschleife

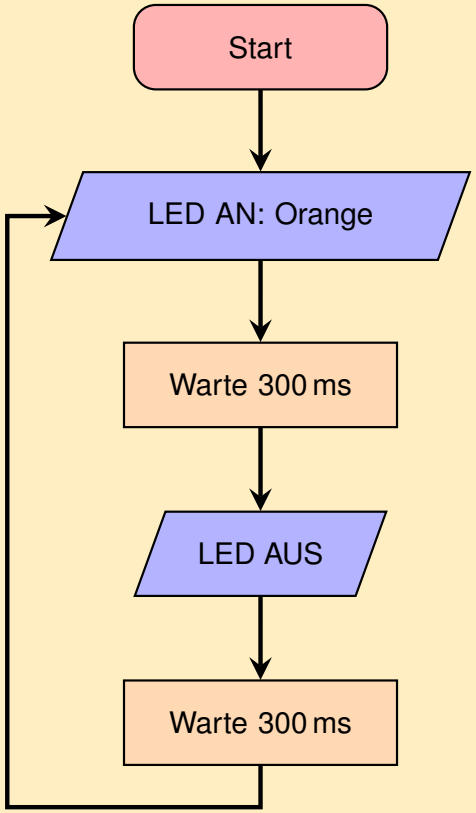


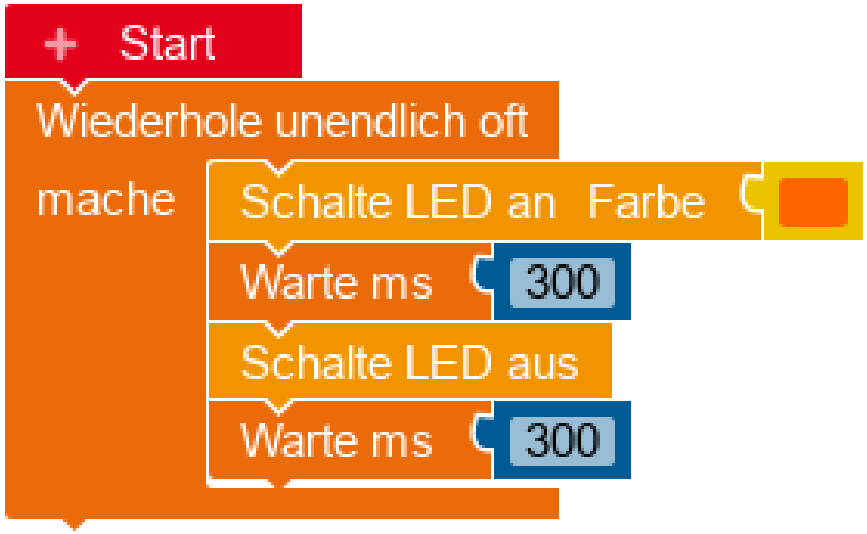
Endlosschleife



Endlosschleife Blinker

Flussdiagramm

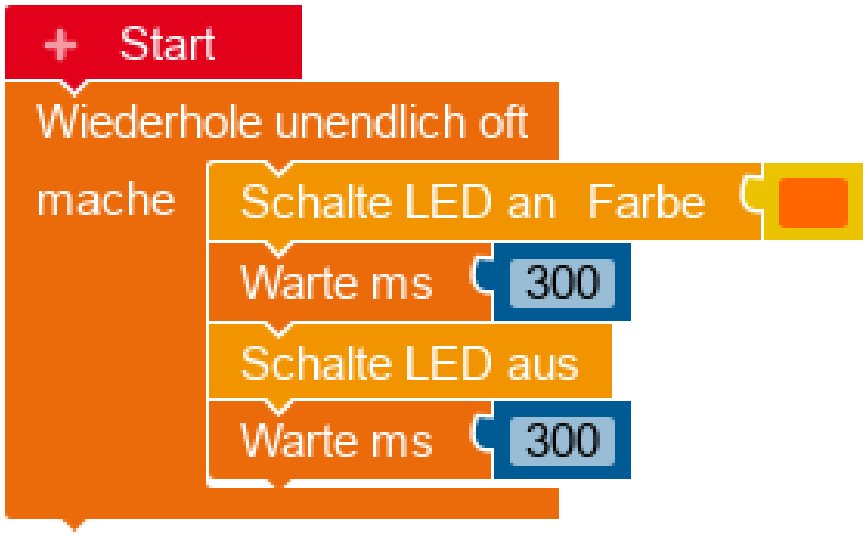




Endlosschleife Blinker

Struktogramm

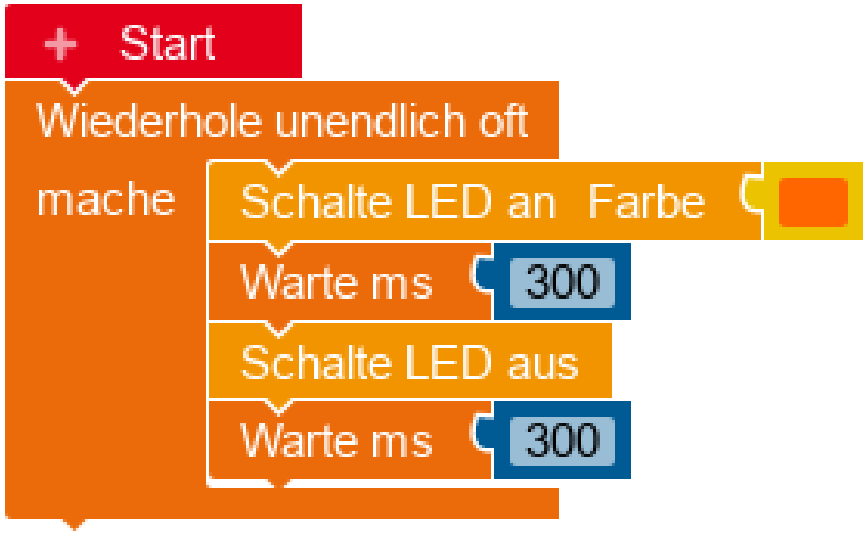
LED AN: Orange
Warte 300 ms
LED AUS
Warte 300 ms



Endlosschleife Blinker

Pseudocode

- START
- Wiederhole unendlich oft
  - Schalte RGB-LED auf Orange
  - Warte 300 ms
  - Schalte RGB-LED aus
  - Warte 300 ms
- STOP



Endlosschleife Blinker

Aufgabe 1

Die RGB-LED soll als Blaulicht, ähnlich wie bei einem modernen Feuerwehrauto, arbeiten. Dazu soll die LED für 50 ms aufleuchten mit einer Pause von 100 ms dazwischen. Nach jedem dritten Leuchten soll die Pause 400 ms betragen.

a) Erstelle einen Pseudocode, ein Flussdiagramm und ein Struktogramm für einen geeigneten Programmablauf.

b) Programmiere das Programm in NEPO und teste es auf dem Calliope.

Zeige Osten

Aufgabe 2

Der Calliope besitzt einen Kompasssensor, der die Richtung als Winkel ausgibt. Osten liegt bei 90 Grad. Der Calliope soll den Sonnenaufgang anzeigen. Damit das besser funktioniert, soll die Richtungsangabe zwischen 80 und 100 Osten anzeigen.

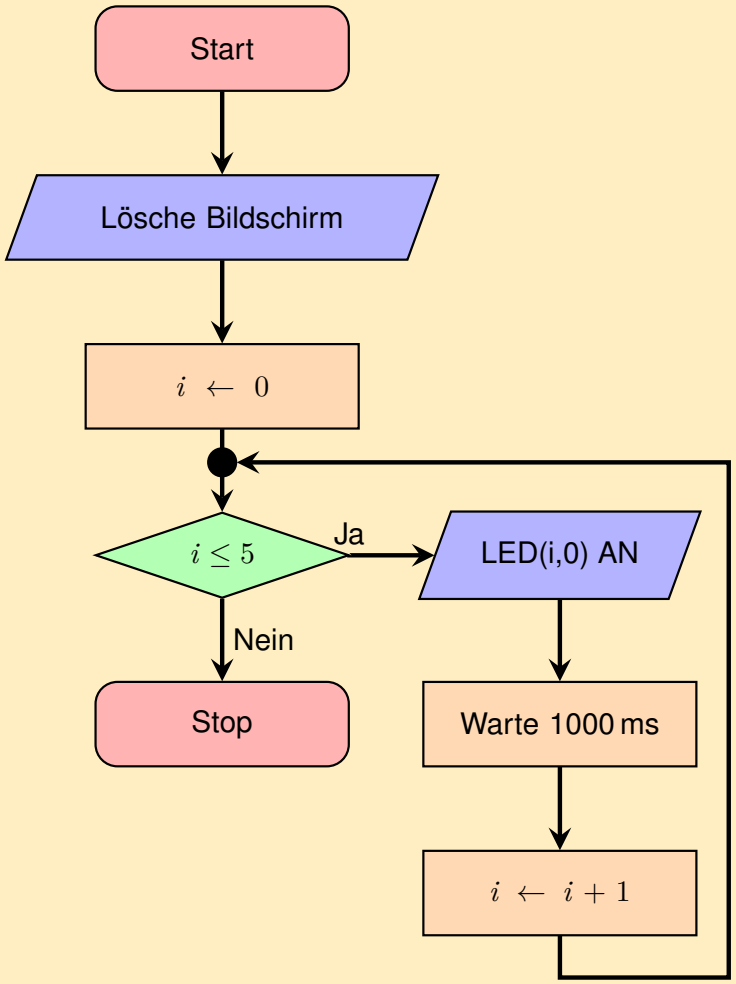
a) Erstelle einen Pseudocode, ein Flussdiagramm und ein Struktogramm für einen geeigneten Programmablauf.

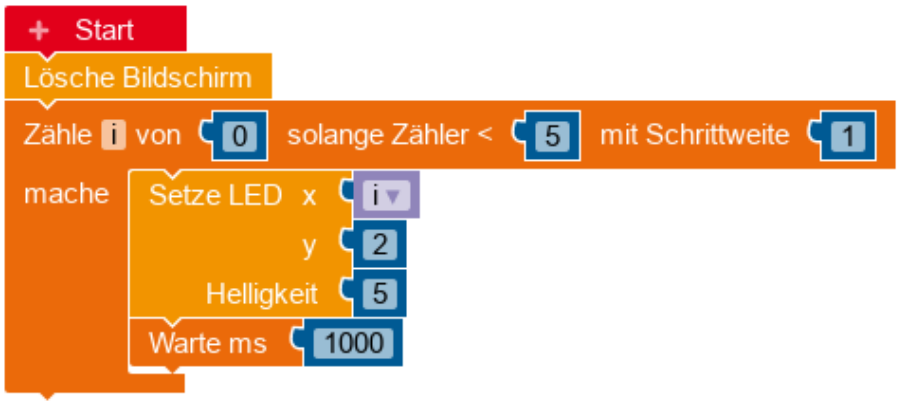
b) Programmiere das Programm in NEPO und teste es auf dem Calliope.

Zählschleife

Zählschleife: LED Lauflicht

Flussdiagramm

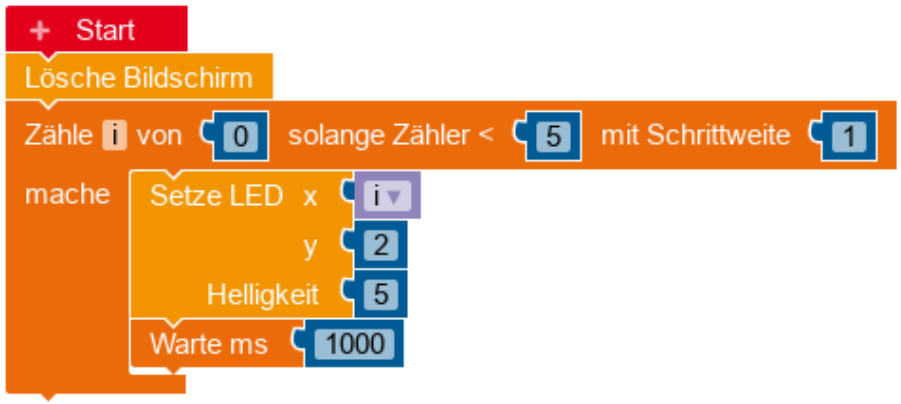




Zählschleife: LED Lauflicht

Struktogramm

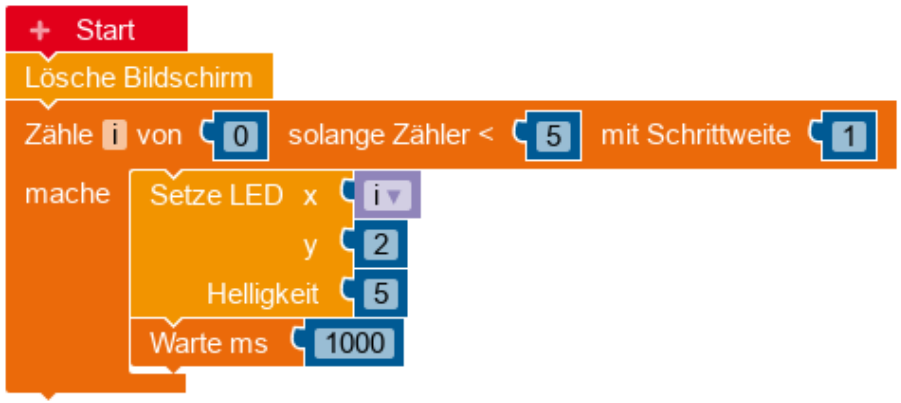
Lösche Bildschirm
$i \leftarrow 0$
$i \leq 5$
LED(i,0) AN
Warte 1000 ms
$i \leftarrow i + 1$



Zählschleife: LED Lauflicht

Pseudocode

- START
- Lösche Bildschirm.
- Setze i auf 0.
- Wiederhole solange i kleiner gleich 5 ist.
  - LED an der Stelle (i,0) anschalten.
  - Warte 1000 ms.
  - Erhöhe i um 1.
- STOP



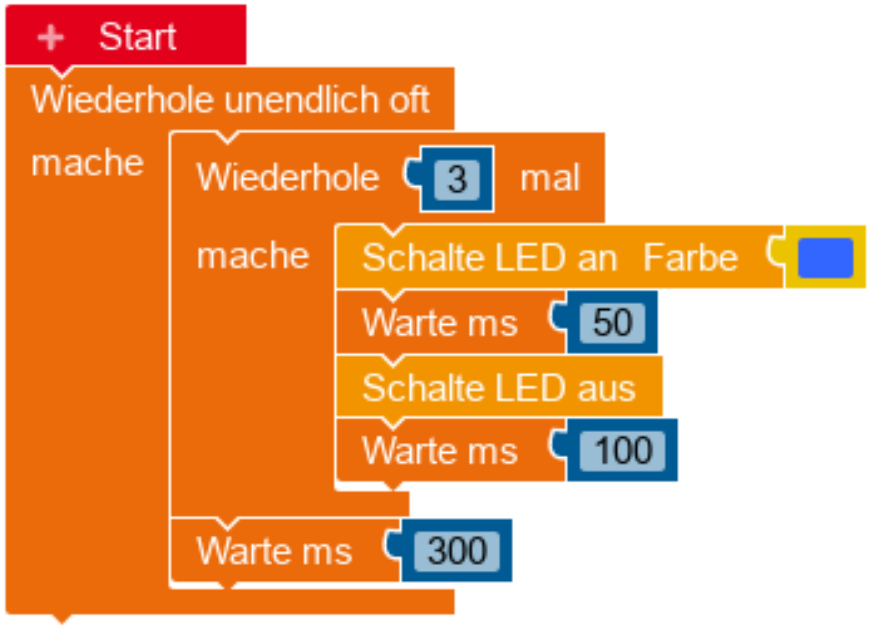
Zählschleife

Aufgabe 3

Gegeben ist das rechts stehende Programm.

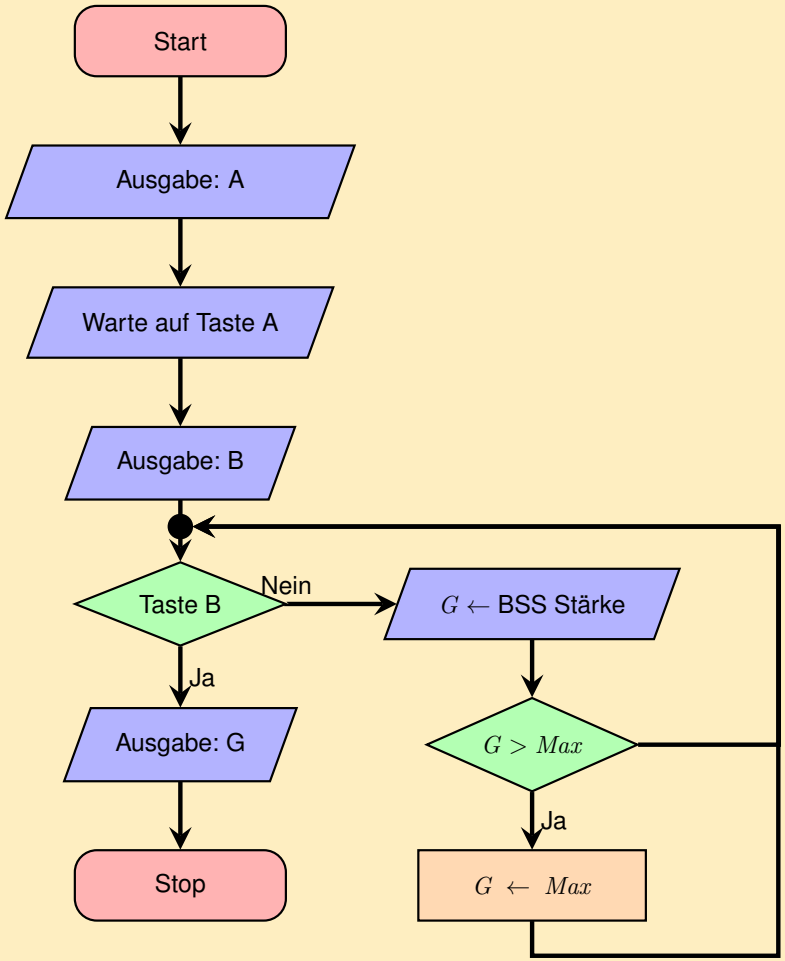
a) Beschreibe die Funktion des Programms.

b) Erstelle Flussdiagramm, Struktogramm und Pseudocode für dieses Programm.



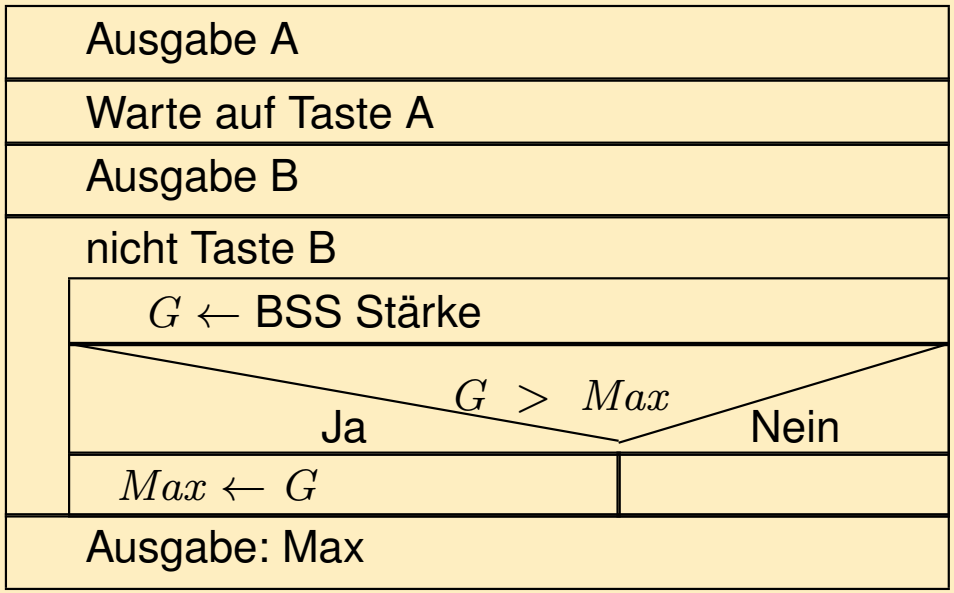
Solange-Schleife: Maximale Beschleunigung

Flussdiagramm



Solange-Schleife: Maximale Beschleunigung

Struktogramm



Solange-Schleife: Maximale Beschleunigung

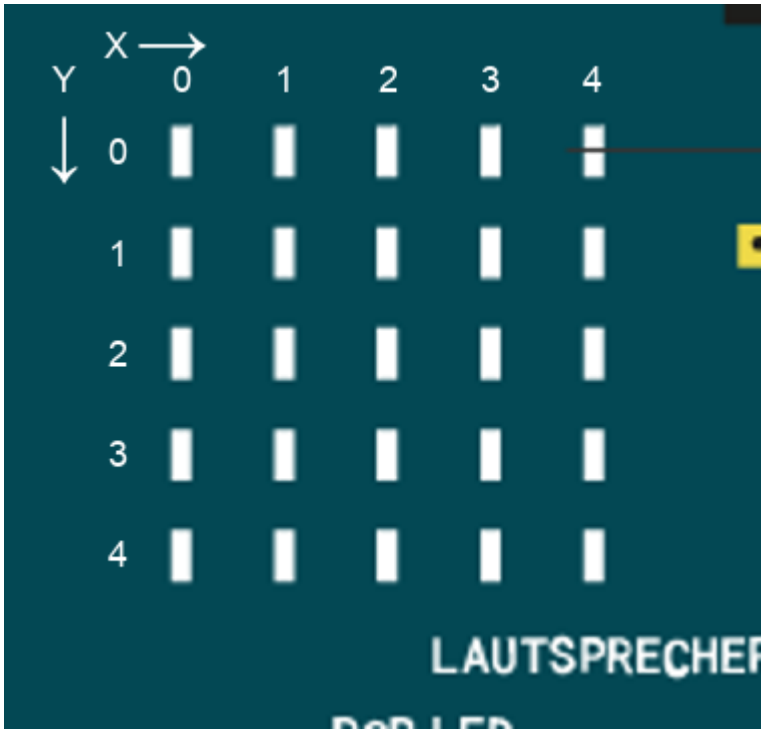
Pseudocode

- START
- Ausgabe ’A’.
- Warte auf Taste A.
- Schreibe ’B’.
- Wiederhole bis Taste B gedrückt.
  - Lese die Stärke des Beschleunigungssensors aus und speichere in G.
  - Wenn G größer als Max ist, dann
    - Schreibe in Max den Wert von G.
- Ausgabe Max.
- STOP

LED-Anzeige direkt ansteuern

Mit dem Befehl `Setze LED` kann jede einzelne LED der LED-Anzeige angesteuert werden. Dabei kann die Helligkeit von 0 (dunkel) bis 5 (sehr hell) eingestellt werden. Als Parameter erwartet er die Spalte (x) und die Zeile (y) in der sich die LED befindet. Dazu noch die Helligkeit, mit der sie leuchten soll.

Die linke obere Ecke hätte damit die Koordinaten (0,0) und die rechte untere Ecke die Koordinaten (4,4). Die linke untere Ecke erreicht man mit (0,4) und die rechte obere Ecke mit (4,0).



LED-Anzeige direkt ansteuern

Aufgabe 4

Steuere die LEDs mit dem Befehl `Setze LED` möglichst geschickt an.

a) Lasse die mittlere LED aufleuchten.
b) Lasse alle LEDs der untersten Reihe leuchten.
c) Lasse alle LEDs der rechten Reihe leuchten.
d) Lasse die mittlere LED im Sekundentakt blinken.

Beschleunigungssensor

Der im Calliope Mini eingebaute Beschleunigungssensor gibt die gemessene Beschleunigung in tausendstel g zurück, wobei g für die durchschnittliche Erdbeschleunigung von $9{,}81\,\mathrm{m/s^2}$ steht. Dabei gibt es die Möglichkeit, die allgemeine Stärke der Beschleunigung auszugeben, bzw. die Beschleunigung entlang der drei Achsen X, Y und Z, die senkrecht aufeinanderstehen. Wird der Calliope nicht beschleunigt, so erfährt er aber dennoch die Erdbeschleunigung. Setzt man die Werte für die drei Achsen in Relation zueinander, dann kann man die Lage des Calliope Minis bestimmen.

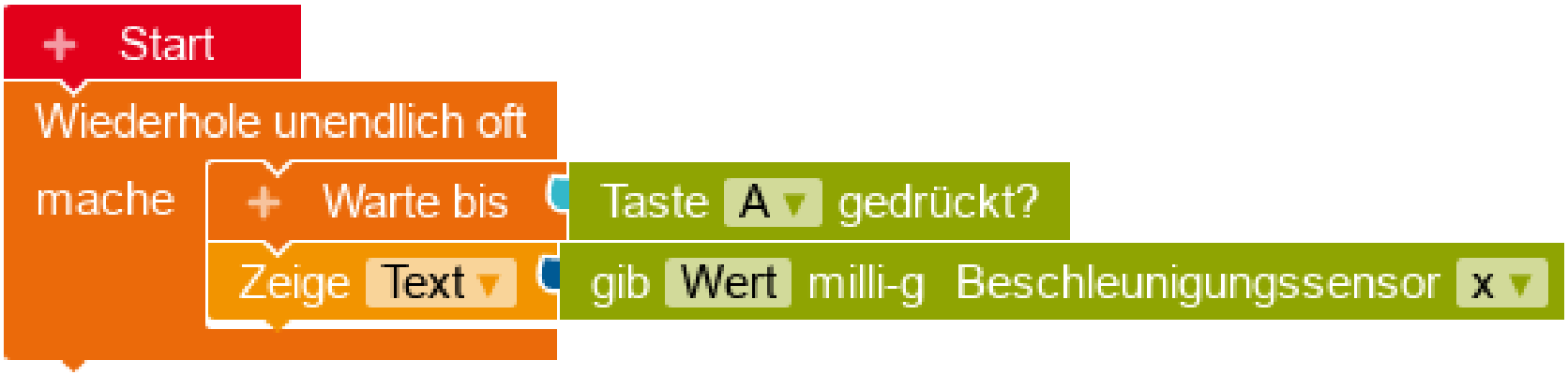
Beschleunigungssensor

Aufgabe 5

Teste mit dem folgenden Programm den Beschleunigungssensor.

a) Bestimme die Lage der X-Achse des Calliopes.

b) Verändere das Programm so, dass auch die Y-Achse und Z-Achse bestimmt werden können.



Beschleunigungssensor

Aufgabe 6

Ein Programm soll entwickelt werden, dass als Bremslicht für Fahrräder dienen kann. Dabei soll die gesamte LED-Anzeige aufleuchten, wenn das Fahrrad abbremst.

a) Überlege von welcher Achse die Beschleunigungswerte ausgelesen werden sollen. Der Grenzwert, bei dem die Leuchte angeht, muss durch ausprobieren ermittelt werden.

b) Erstelle einen Pseudocode und ein Struktogramm für einen geeigneten Programmablauf.

c) Programmiere das Programm in NEPO und teste es auf dem Calliope.

Beschleunigungssensor

Aufgabe 7

Ein Programm soll entwickelt werden, dass die Lage des Calliope Minis anzeigt. Die Anforderungen sind:

- Wenn der Calliope waagerecht liegt, soll die mittlere LED leuchten.
- Wird der Calliope leicht geneigt, solle die entsprechendene zur Mitte benachbarte LED in Richtung der Neigung leuchten.
- Wird der Calliope stark geneigt, dann sollen entsprechend der Neigung eine der äußersten LEDs leuchten.

a) Erstelle einen Pseudocode und ein Struktogramm für einen geeigneten Programmablauf.

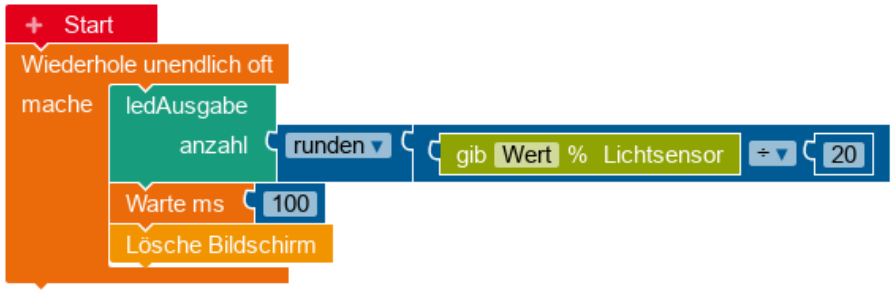
b) Programmiere das Programm in NEPO und teste es auf dem Calliope.

Funktionen: Neue Befehle gestalten

Struktogramm

Hauptprogramm

$N \leftarrow$ Gerundet (Helligkeit in %/20)
Zeige_ N_LEDs(N)
Warte 100 ms
Lösche alle LEDs

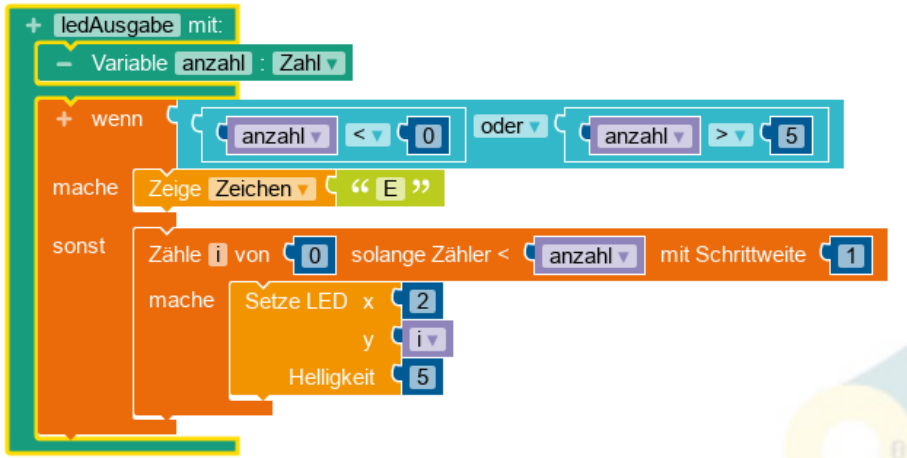


Funktionen: Neue Befehle gestalten

Struktogramm

Funktion: Zeige_N_LEDs(Anzahl)

<table>
<tr><th>$Anzahl < 0$ oder $Anzahl > 5$
Ja</th><th>Nein</th></tr>
<tr><td>Fehlermeldung ausgeben</td><td>Zähle i von 0 um 1 hoch solange $i < Anzahl$ ist
<table><tr><td>LED(2,i) anschalten</td></tr></table></td></tr>
<tr><td>$\varnothing$</td><td></td></tr>
</table>



Funktionen: Variablen

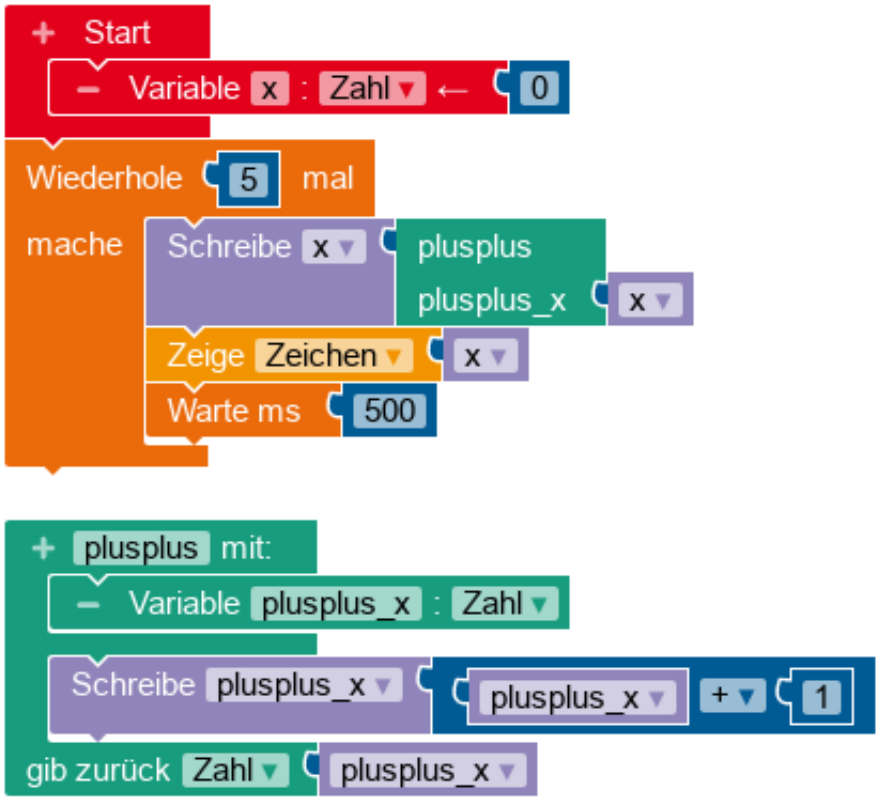
Lokale Variablen

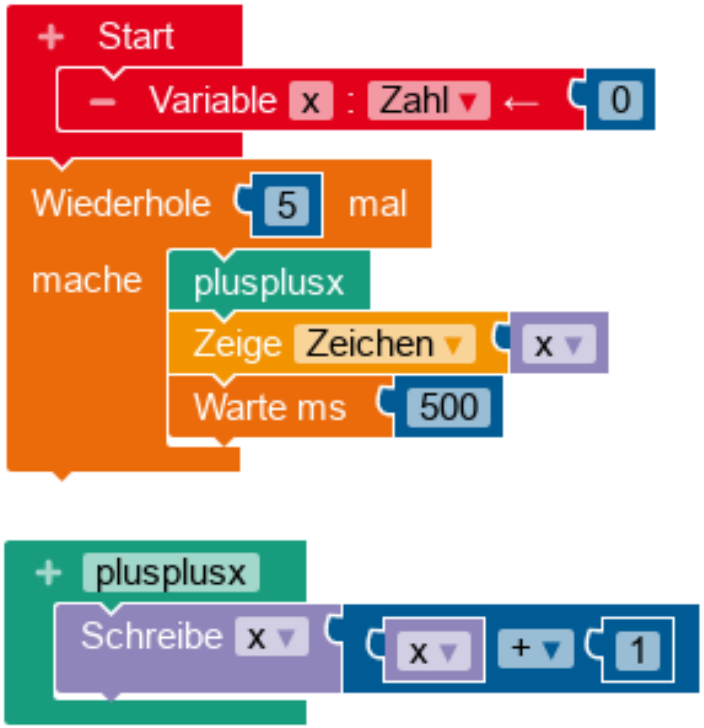
Variablen, die in einer Funktion deklariert worden sind, sind nur in dieser Funktion nutzbar. Sie werden als **lokale Variablen** bezeichnet.

Globale Variablen

Die im Start-Block deklarierten Variablen stehen auch in Funktionen zur Verfügung. Sie werden als **globale Variablen** bezeichnet.

Funktionen: Variablen





Funktionen: LED Anzeige

Aufgabe 8

Die Funktion `ledAusgabe` soll so verbessert werden, dass nun die Werte 0 bis 25 durch leuchtende LEDs dargestellt werden können.

a) Erstelle einen Pseudocode und ein Struktogramm für einen geeigneten Programmablauf.

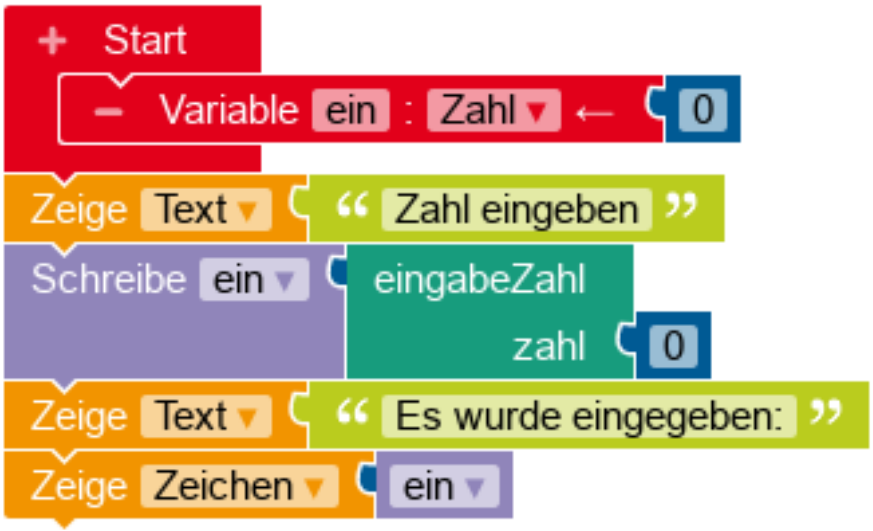
b) Programmiere das Programm in NEPO und teste es auf dem Calliope. Entwickle dazu ein geeignetes Hauptprogramm.

Funktionen mit Rückgabewert

Struktogramm

Hauptprogramm

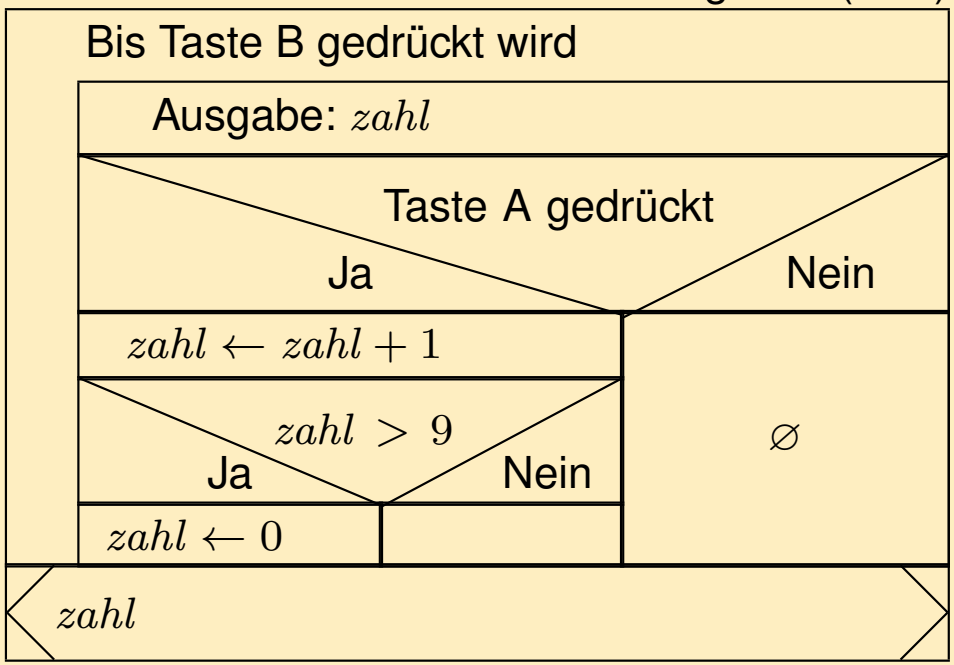
Ausgabe: Zahl eingeben
$ein \leftarrow$ Zahl zwischen 0 und 9 eingeben(0)
Ausgabe: Es wurde eingegeben:
Ausgabe: ein

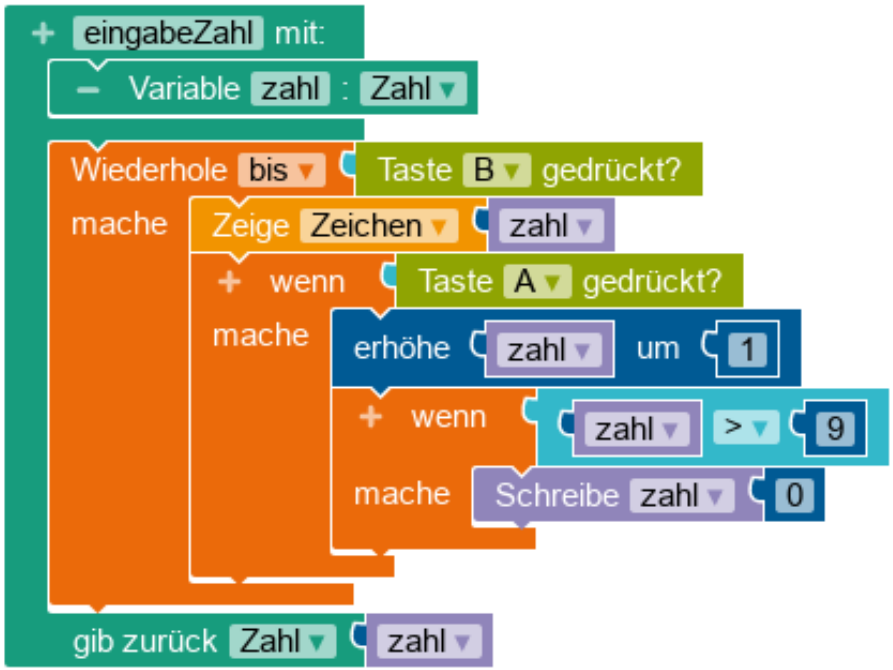


Funktionen mit Rückgabewert

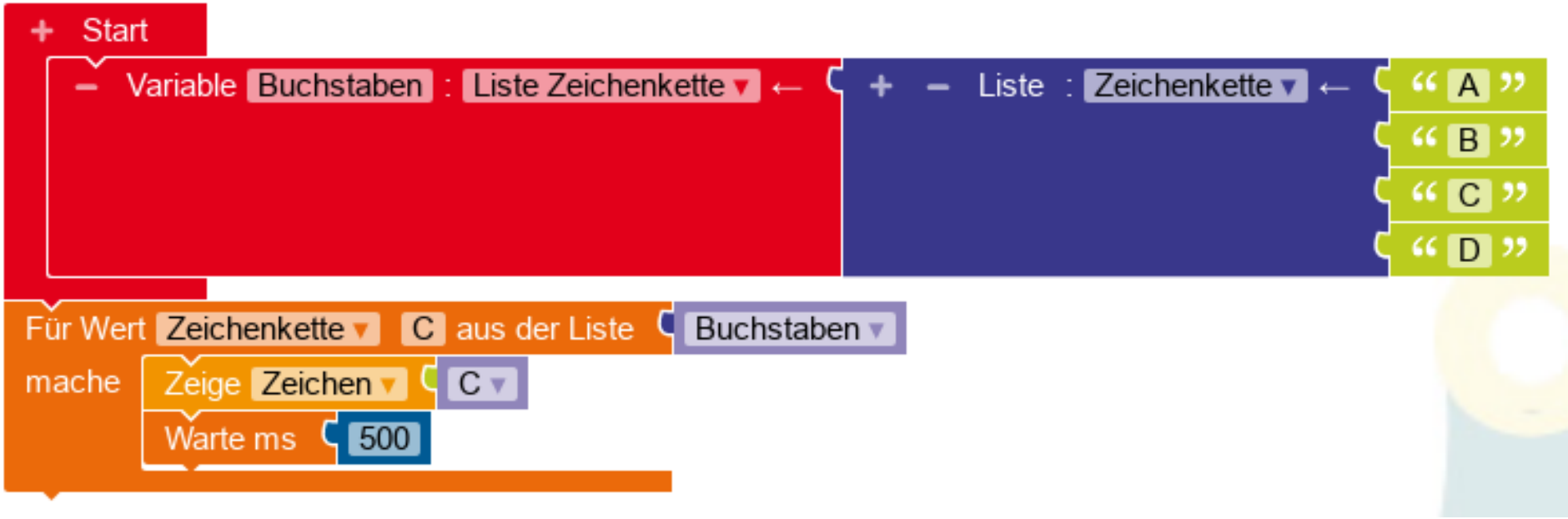
Struktogramm

Funktion: Zahl zwischen 0 und 9 eingeben (zahl)

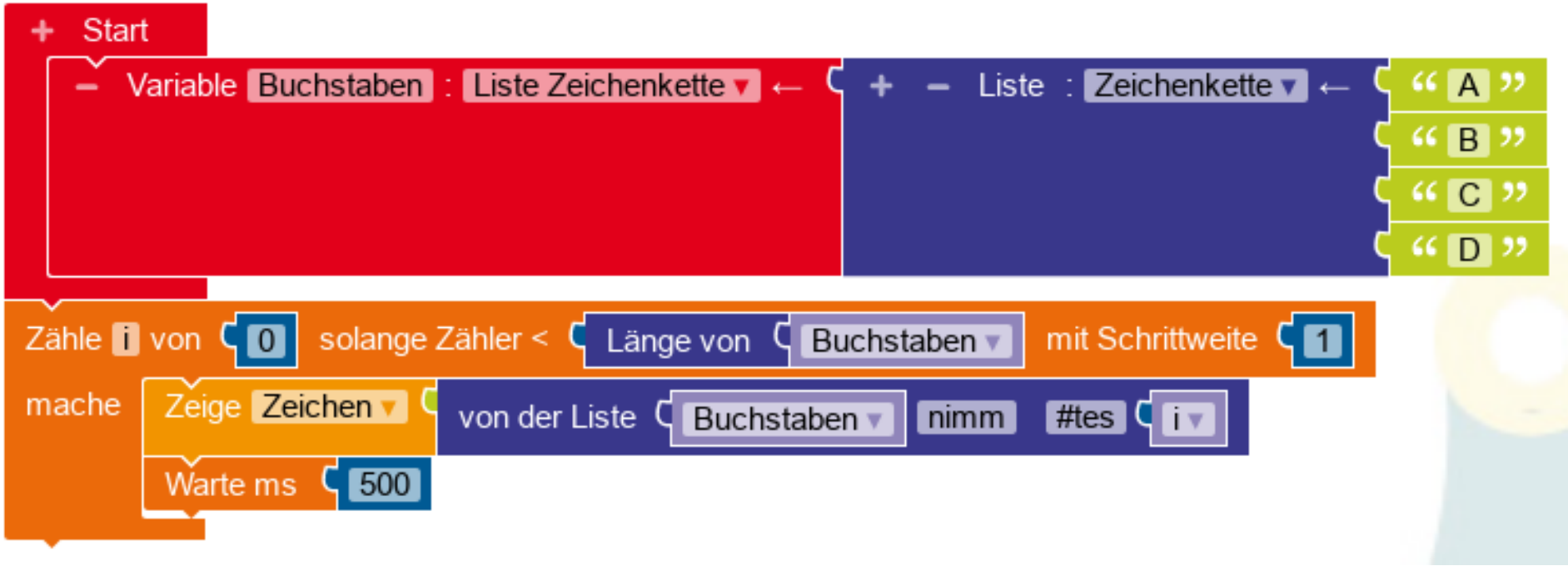




Listen


Start
Variable Buchstaben : Liste Zeichenkette ←
Liste : Zeichenkette ←
A
B
C
D
Für Wert Zeichenkette C aus der Liste Buchstaben
mache
Zeige Zeichen C
Warte ms 500

Listen


Start
Variable Buchstaben : Liste Zeichenkette ←
Liste : Zeichenkette ←
“ A ”
“ B ”
“ C ”
“ D ”
Zähle i von 0 solange Zähler < Länge von Buchstaben mit Schrittweite 1
mache
Zeige Zeichen
von der Liste Buchstaben nimm #tes i
Warte ms 500

Listen

Aufgabe 9

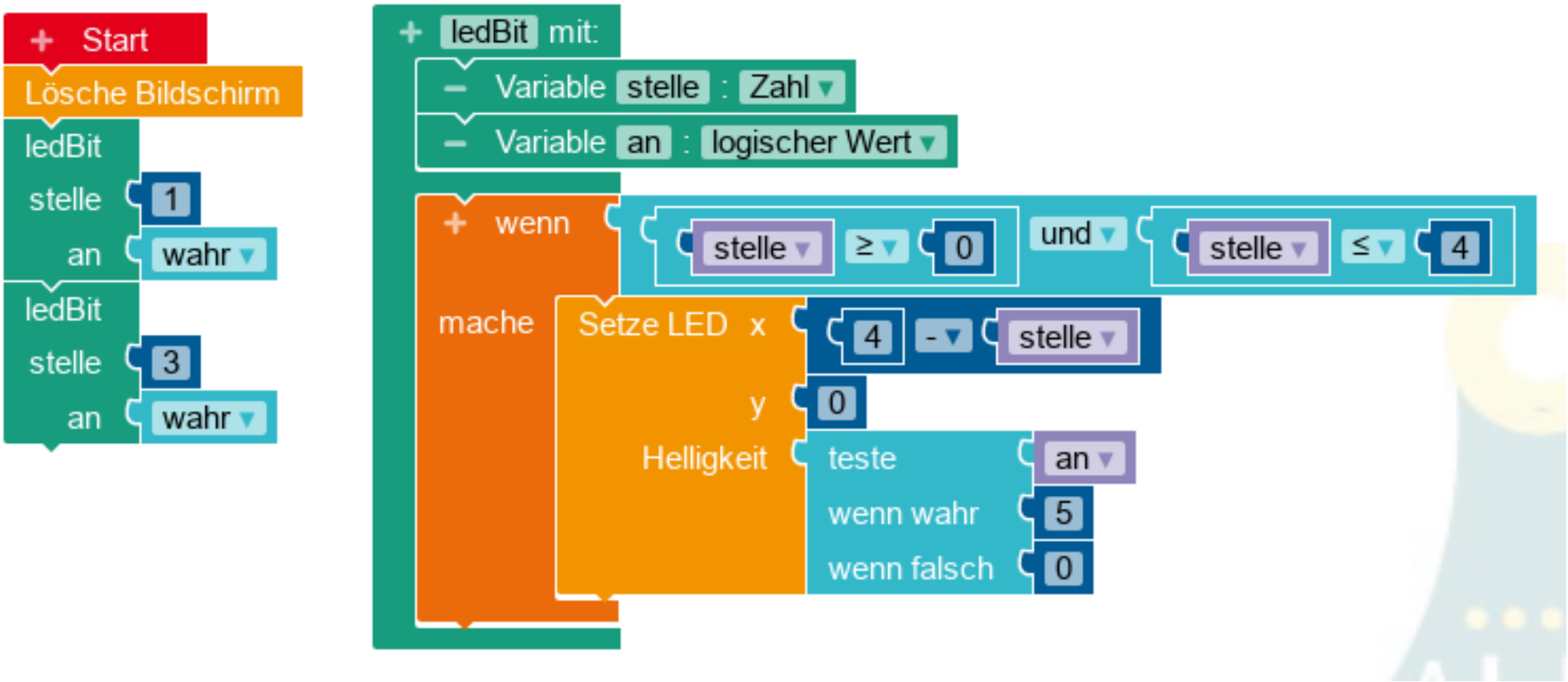
Entwickle ein Programm, dass zufällig einen Buchstaben aus der Liste auf dem Bildschirm ausgibt. Entwerfe ein passendes Struktogramm und schreibe das Programm.

Dezimalzahl zu Nibble

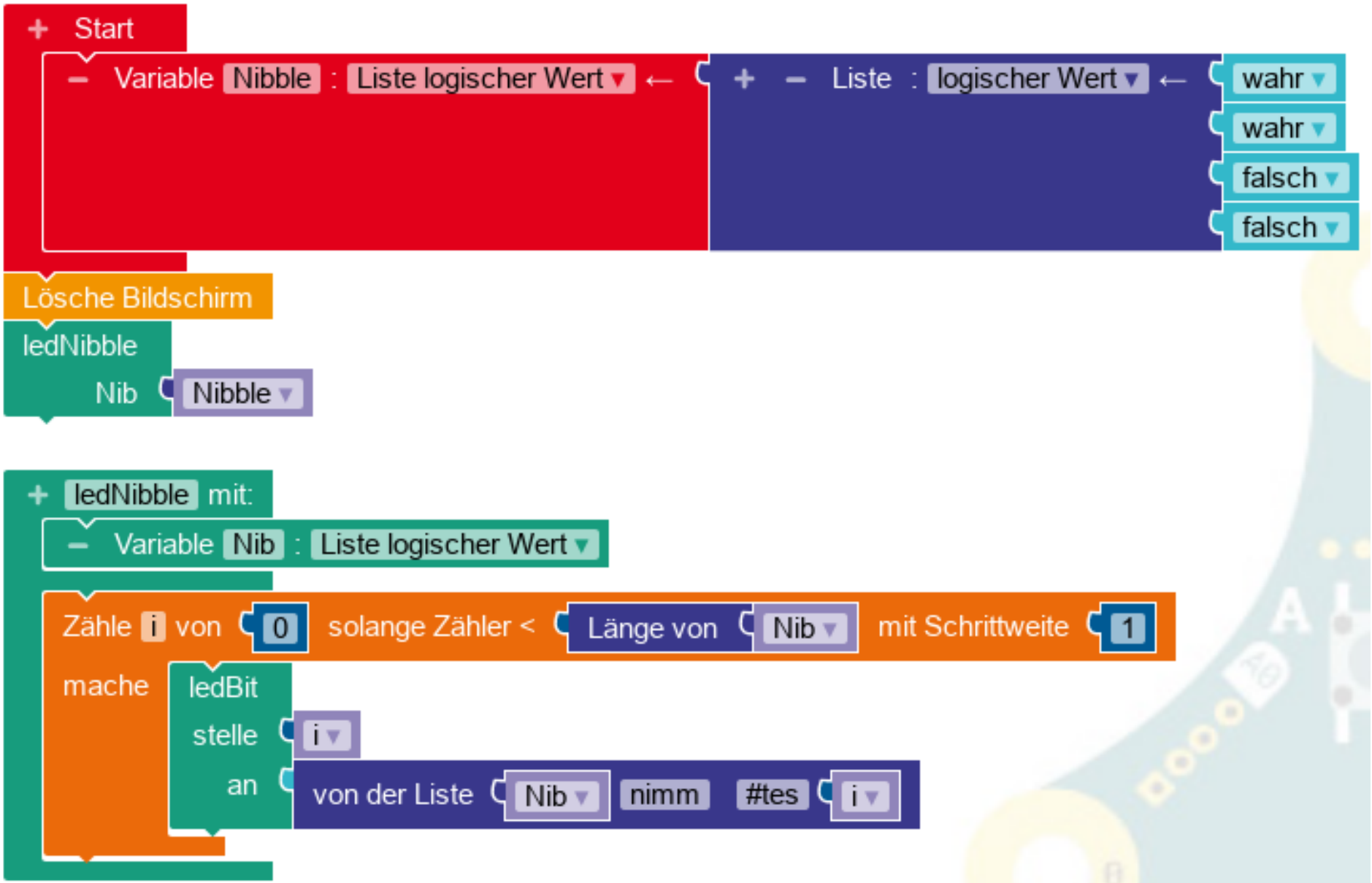
Aufgabe

Eine Dezimalzahl soll als Binärzahl (Nibble) mit den LEDs des Calliope dargestellt werden.

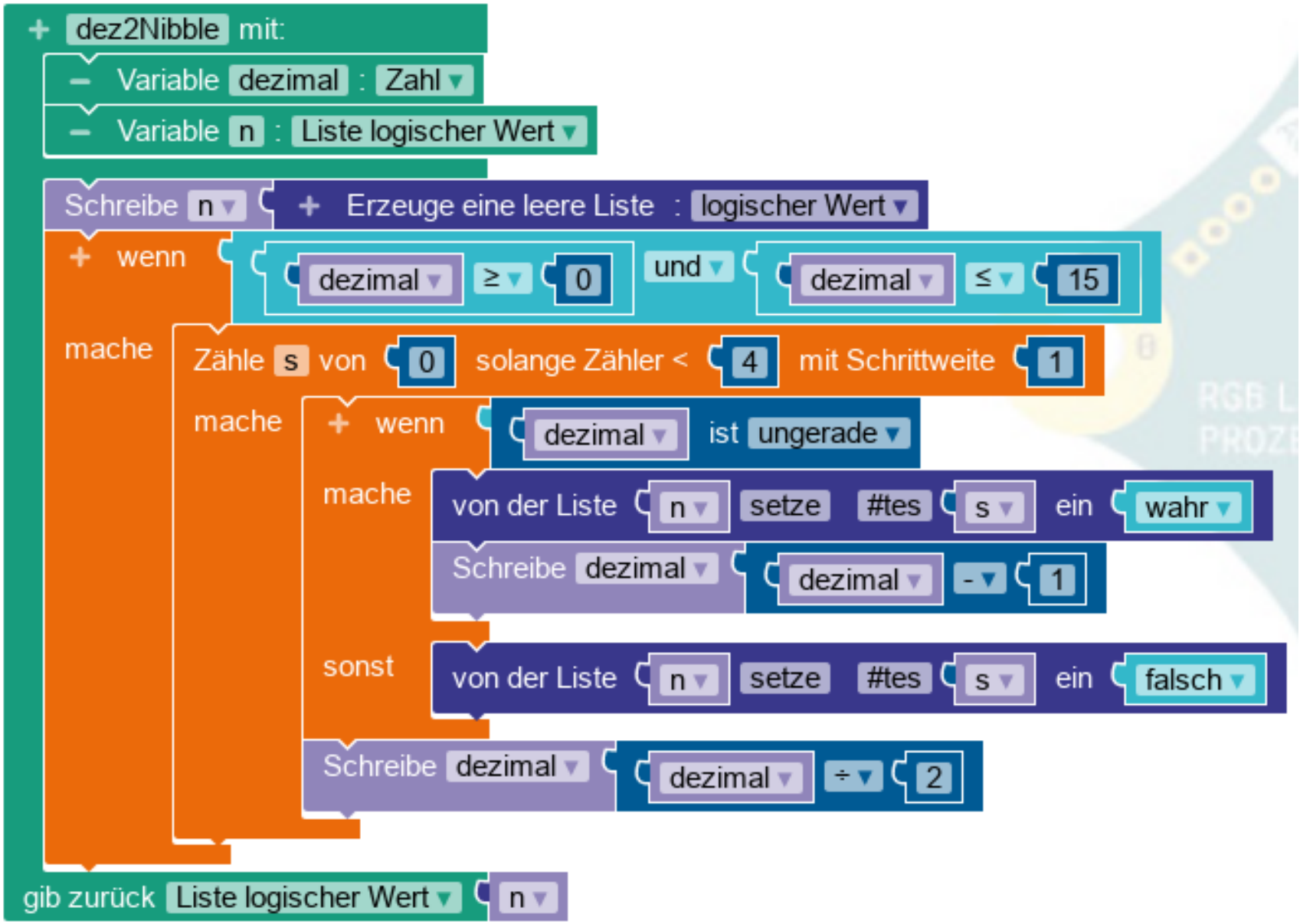
ledBit(stelle, an)



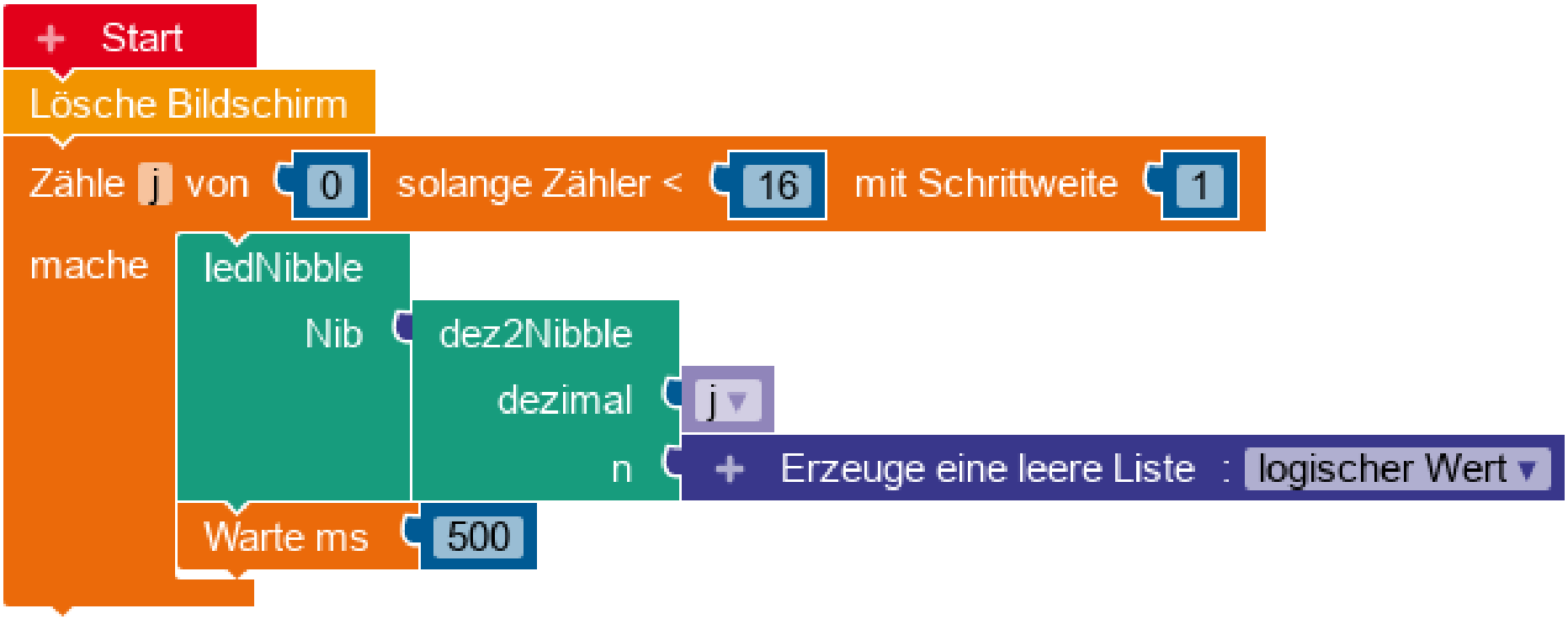
ledNibble(Nib)



dez2Nibble(dezimal, n)



Testprogramm für dez2Nibble(dezimal, n)



Corona-Zähler

- Innerhalb eines Geschäftes dürfen sich nur 16 Personen aufhalten.
- Am Eingang steht ein Sicherheitsposten, der diese Bedingung überwacht.
- Überwacht wird dies mit einem Calliope.
  - Drücken der Taste A zählt einen Kunden, der in das Geschäft geht.
  - Drücken der Taste B zählt einen Kunden, der aus dem Geschäft kommt.
  - Die Anzahl der Kunden wird durch die Anzahl der leuchtenden LED angezeigt.
  - Beim Erreichen der Höchstzahl ertönt ein Signalton und die RGB-LED blinkt rot.
  - Über 10 Kunden leuchtet die RGB-LED gelb, sonst leuchtet sie grün.

Corona-Zähler

Aufgabe 10

a) Überlege, welche Variablen benötigt werden, um ein passendes Programm zu realisieren.

b) Erstelle ein Struktogramm für den Algorithmus.

c) Entwickle ein Programm für den Calliope Mini, das den Corona-Zähler realisiert.

d) Überlege, wie man den Corona-Zähler noch verbessern könnte.

Langton-Ameise

- Turing-Maschine
- 1986 von Christopher Langton entwickelt
- Deterministisches – d.h. nicht zufallsbedingtes – System mit einfachen Regeln

Langton-Ameise

Ausgangssituation

- Unendliches Quadratgitterfeld
- Felder können schwarz oder weiß sein.
- Am Anfang sind alle Felder weiß.
- Ameise sitzt auf einem Feld und schaut in Richtung einer der vier Seiten.

Algorithmus

1. Auf einem weißen Feld drehe 90 Grad nach rechts. Auf einem schwarzen Feld drehe 90 Grad nach links.
2. Wechsle die Farbe des Feldes.
3. Gehe zum nächsten Feld in Blickrichtung.

Langton-Ameise

Aufgabe 11

a) Spiele den Algorithmus auf kariertem Papier nach. (mindestens bis Schritt 30)

b) Überlege, welche Variablen benötigt werden, um ein passendes Programm zu realisieren.

c) Erstelle ein Struktogramm für den Algorithmus.

d) Entwickle ein Programm für den Calliope Mini, das die Langton-Ameise auf dem LED-Display darstellt.

e) Überlege, welche Probleme auftreten (könnten).