

MICRO:BIT

Beispiele für den Unterricht



Inhalt

1.) Drahtlose Kommunikation über Bluetooth.....	2
2.) LED Muster anzeigen.....	4
3.) LED Muster wiederholt anzeigen	5
4.) Kompassfunktion testen	6
5.) Kompassfunktion mit Toleranzbereichen	8
6.) Kopf oder Zahl	10
7.) 1 x 1 Trainer.....	11
8.) Temperaturen messen	13
9.) Countdown.....	14
10.) Countdown - Erweiterung:.....	15
11.) Schrittzähler	16
12.) QUIZ Fragen.....	17
13.) Einfaches Addieren	19
14.) Schere – Stein - Papier	20
15.) Smileys mit dem Micro:Bit anzeigen.....	22
16.) „Freude schöner Götterfunken“	23
17.) Zeig mir die Temperatur.....	24

1.) Drahtlose Kommunikation über Bluetooth

Autoren: Johann Martin/Rainsperger Herbert

Schwierigkeitsgrad: Mittel

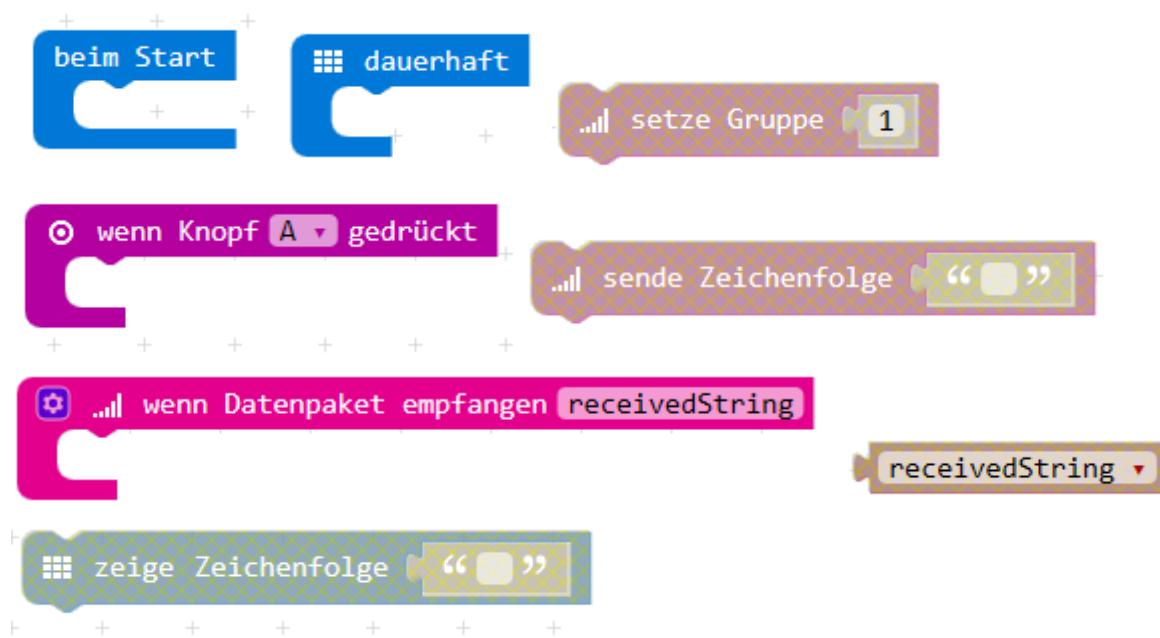
Beschreibung der Aufgabenstellung:

Micro:Bits können über Bluetooth miteinander kommunizieren. Mit Hilfe geeigneter Blöcke soll durch Drücken der Taste A oder B ein Text auf den zweiten Micro:Bit übertragen werden.

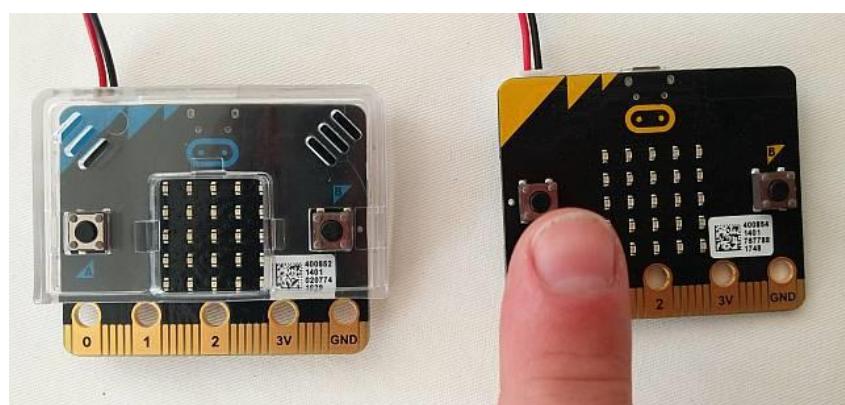
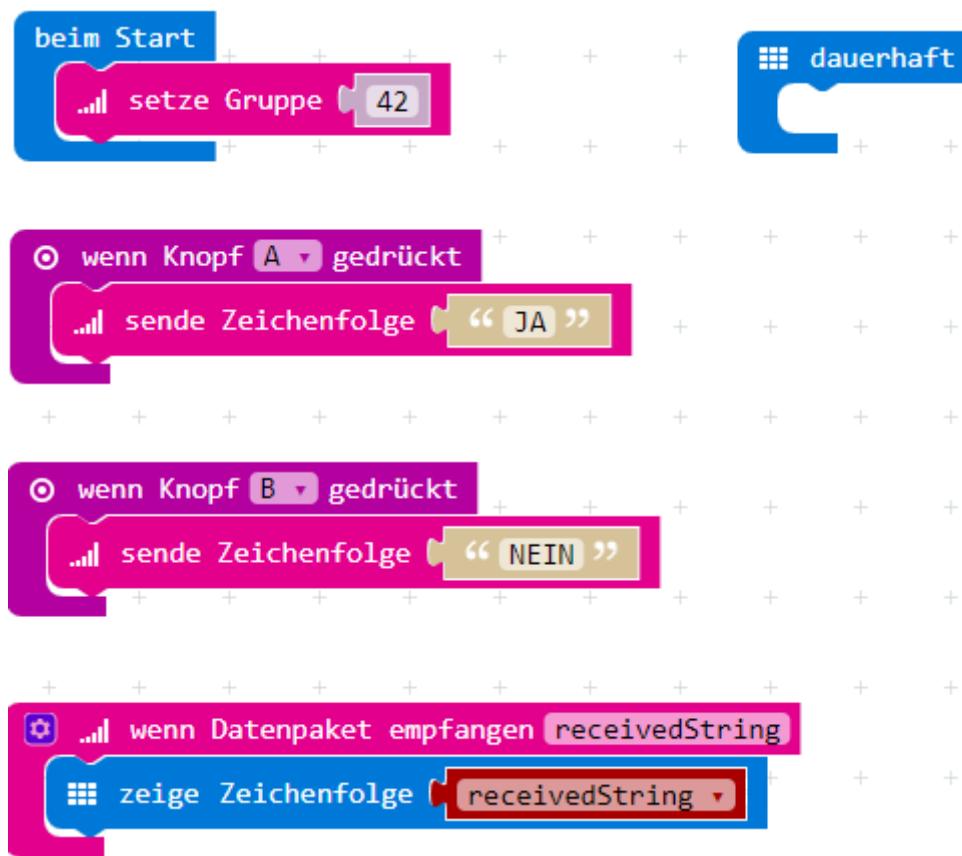
Was wird gelernt?

- Festlegen einer Funkgruppe (alle micro:bits, die dieser Funkgruppe angehören, können Nachrichten senden und empfangen)
- Sendezeichen für die Tasten festlegen
- Senden und Empfangen von Texten über Bluetooth

Welche Blockbefehle werden verwendet?



Lösung:



2.) LED Muster anzeigen

AutorInnen: Christine Gräßl, Erika Knellwolf, Franz Borotschnig

Schwierigkeitsgrad: Anfänger

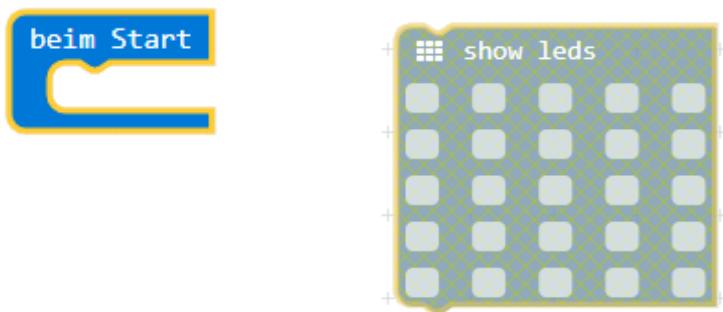
Beschreibung der Aufgabenstellung: Darstellung unterschiedlicher LED Muster.

Was wird gelernt?

- Arbeiten mit Programmierblöcken
- Ein Micro:Bit - entwickeltes Programm starten.

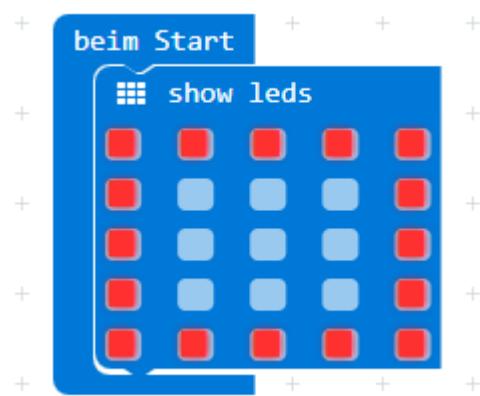
Welche Programmblöcke werden verwendet?

Blockbefehle: Grundlagen:



Lösung:

Screenshot des Programms (Block):



3.) LED Muster wiederholt anzeigen

AutorInnen: Christine Gräßl, Erika Knellwolf, Franz Borotschnig

Schwierigkeitsgrad (Anfänger, Mittel, Fortgeschritten): Anfänger

Beschreibung der Aufgabenstellung:

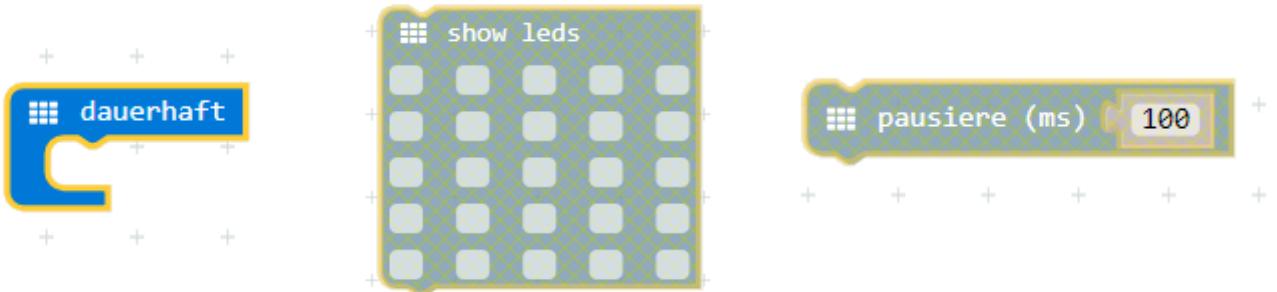
Darstellung unterschiedlicher LED Muster in einer dauerhaften Schleife.

Was wird gelernt?

- Arbeiten mit Programmierblöcken
- Ein Micro:Bit entwickeltes Programm starten.

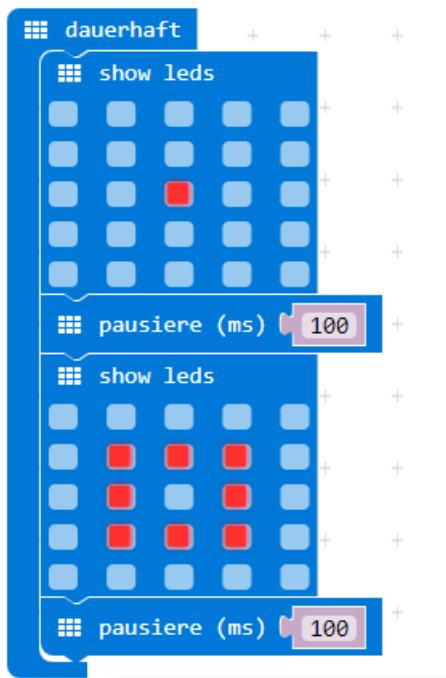
Welche Programmblöcke werden verwendet?

Blockbefehle: Grundlagen



Lösung:

Screenshot des Programms (Block):



4.) Kompassfunktion testen

Autoren: Johann Martin/Rainsperger Herbert

Schwierigkeitsgrad : Mittel

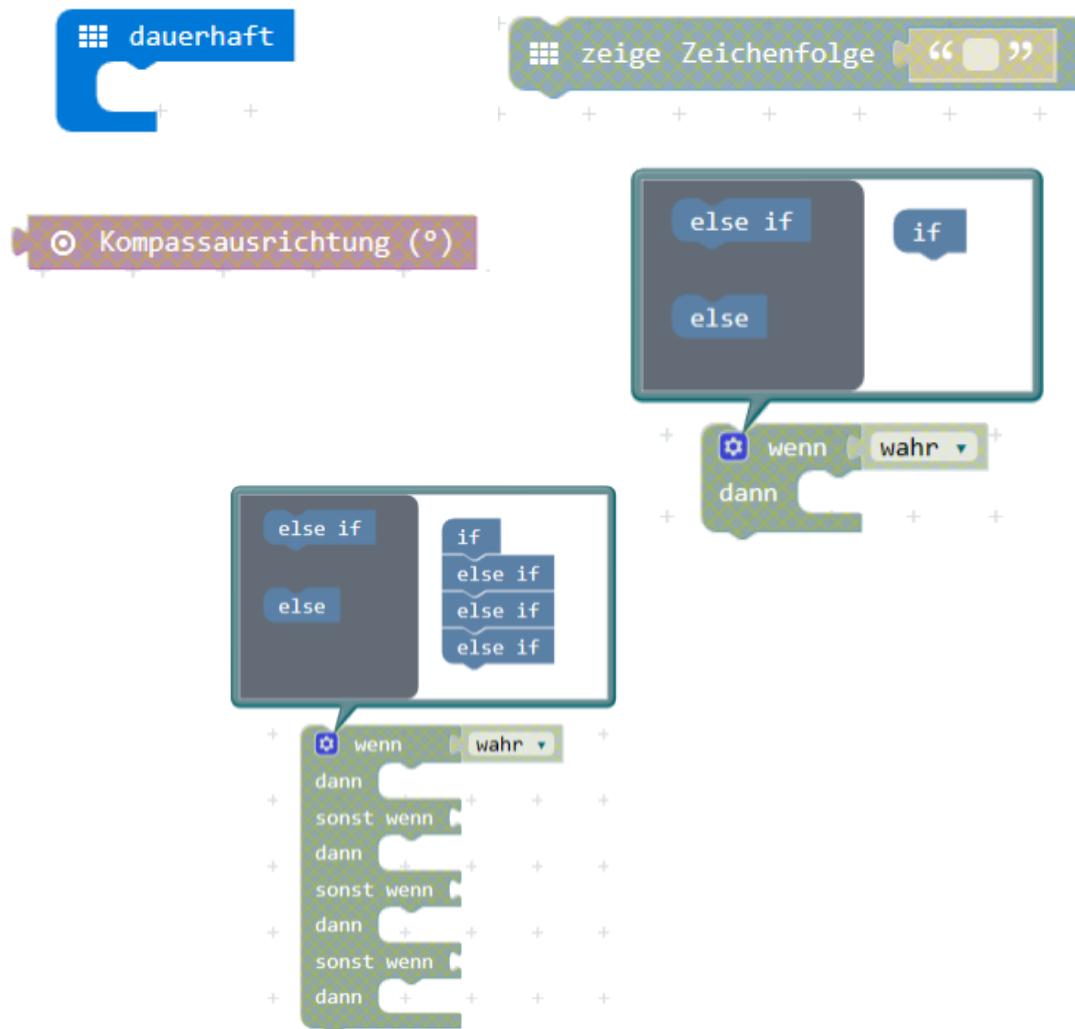
Beschreibung der Aufgabenstellung:

Micro:Bits haben einen Magnetfeldsensor, mit dem die Nordrichtung bestimmt werden kann. Mit Hilfe geeigneter Blöcke und unter Verwendung des Magnetfeldsensors kann der Micro:Bit die vier Himmelsrichtungen anzeigen.

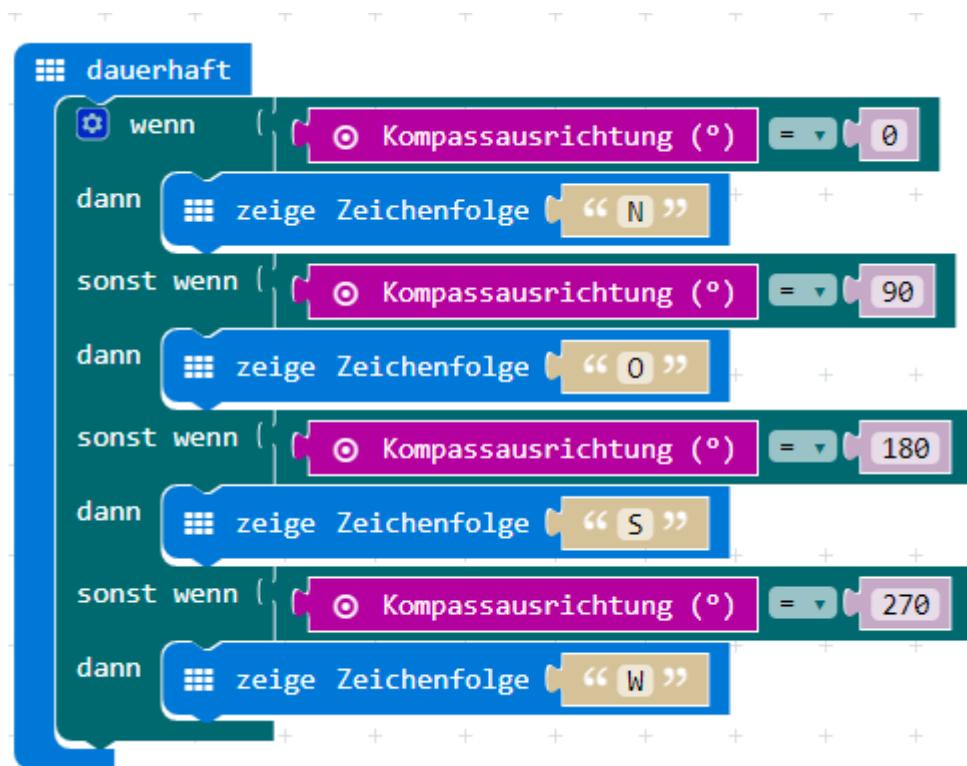
Was wird gelernt?

- Verwenden von logische Funktionen (Wenn, dann, sonst)
 - Bestimmen der Kompassausrichtung
 - Kalibrieren des Kompasses bei der ersten Verwendung

Welche Blockbefehle werden verwendet?



Lösung:



5.) Kompassfunktion mit Toleranzbereichen

Autoren: Johann Martin/Rainsperger Herbert

Schwierigkeitsgrad: Mittel

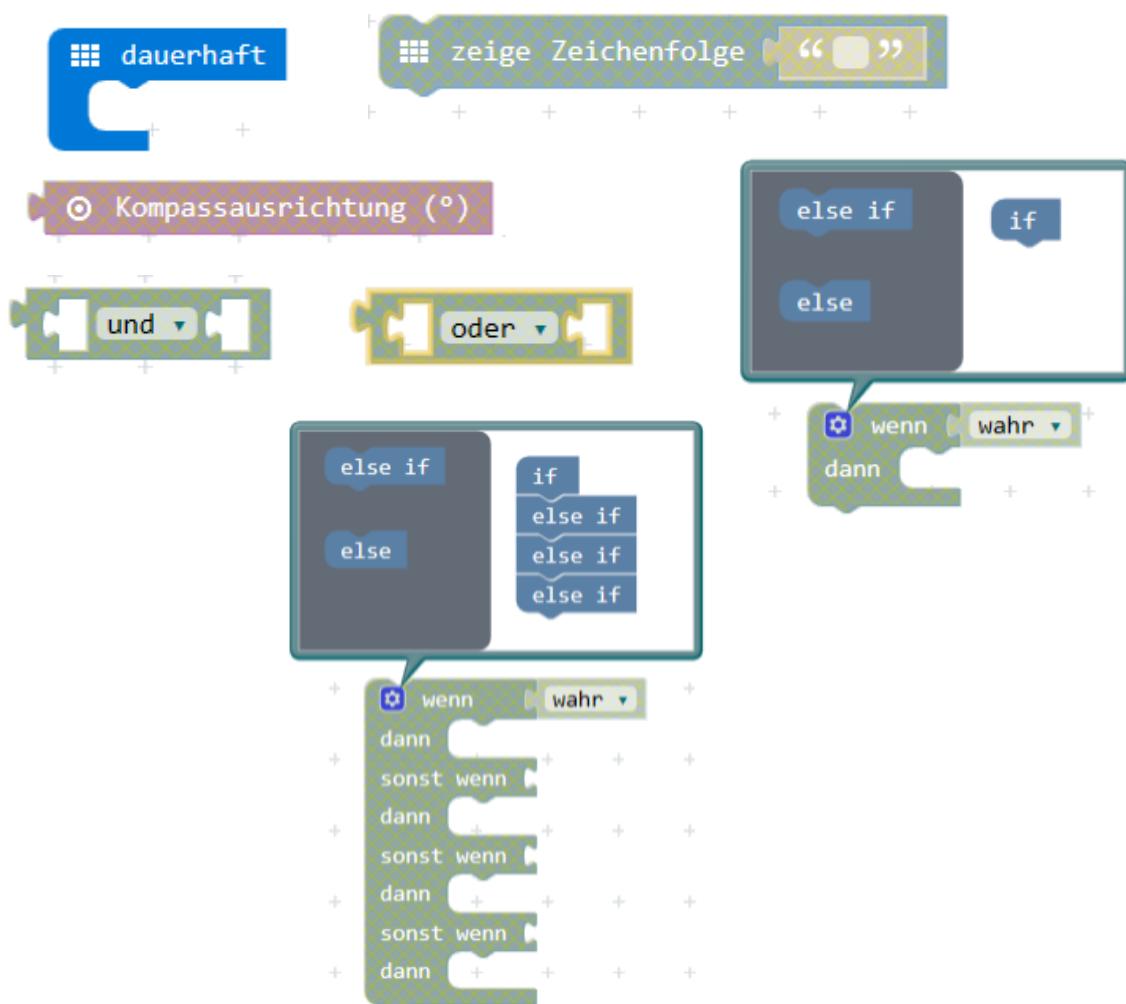
Beschreibung der Aufgabenstellung:

Micro:Bits haben einen Magnetfeldsensor, mit dem die Nordrichtung bestimmt werden kann. Mit Hilfe geeigneter Blöcke und unter Verwendung des Magnetfeldsensors kann der Micro:Bit die vier Himmelsrichtungen anzeigen. Für Norden, Süden, Westen und Osten muss eine Abweichung von 20° pro Himmelsrichtung vorgesehen werden.

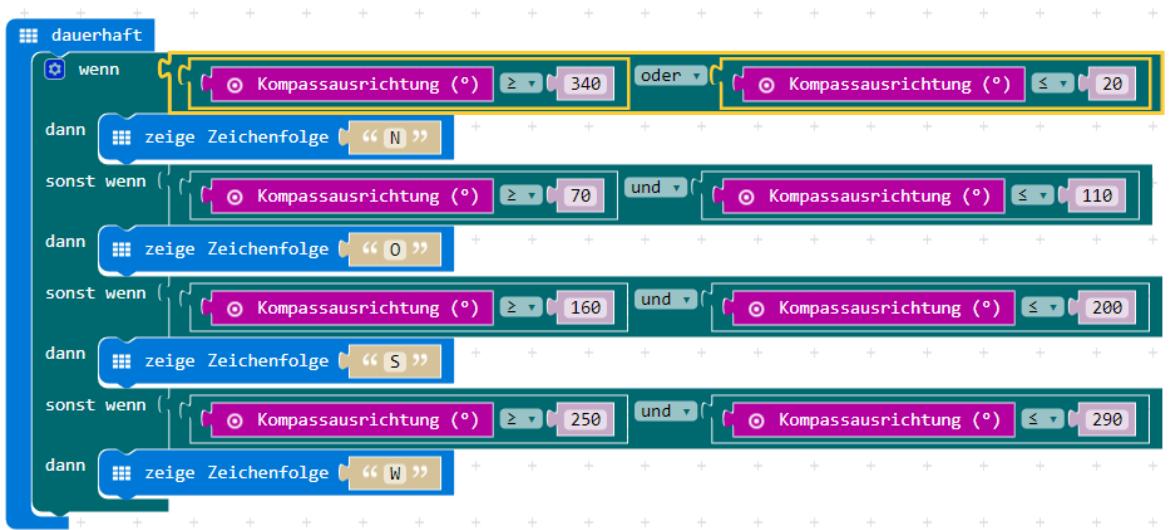
Was wird gelernt?

- Verwenden von logische Funktionen (Wenn, dann, sonst, und, oder)
- Bestimmen der Kompassausrichtung
- Kalibrieren des Kompasses bei der ersten Verwendung

Welche Blockbefehle werden verwendet?



Lösung:



6.) Kopf oder Zahl

Autor: Philip Drußnitzer

Schwierigkeitsgrad: Anfänger

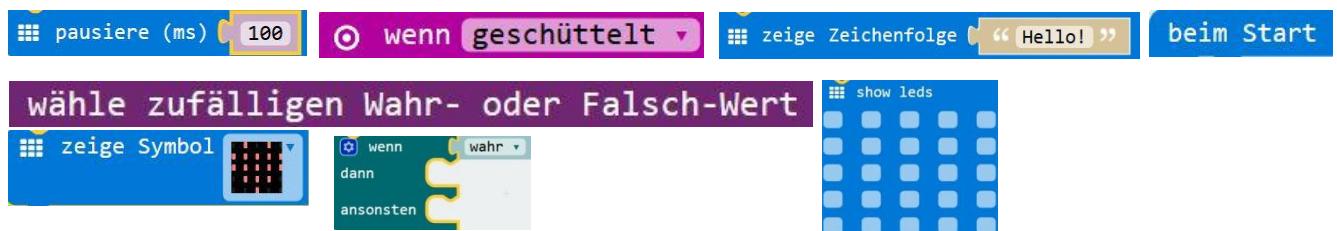
Beschreibung der Aufgabenstellung:

Programmiere einen Generator, der zufällig Kopf oder Zahl ausgibt. Das Programm soll erst nach dem Anzeigen eines Starttextes und nach dem Schütteln des Micro:Bits starten. Zuvor soll der Micro:Bit eine kleine Animation abspielen.

Was wird gelernt?

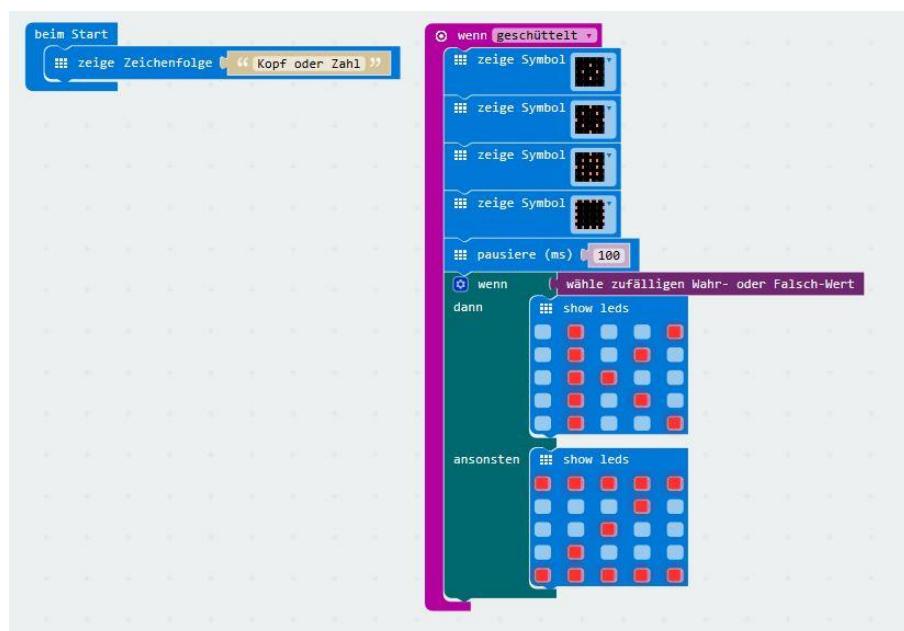
- Anzeigen eines Textes beim Start
- Konfigurieren des Bewegungssensors
- Erstellen eines Zufallsgenerators
- Konfigurieren der LEDs

Welche Programmblöcke werden verwendet?



Lösung:

Screenshot des Programms (Block):



7.) 1 x 1 Trainer

Autorin: Gräßl Christina

Schwierigkeitsgrad : Fortgeschritten

Beschreibung der Aufgabenstellung:

Erstelle einen 1 x 1 Trainer. Wenn du auf den Knopf A des Microbit drückst, generiert er eine Multiplikation von zwei Zufallszahlen. Jede dieser Zufallszahlen soll eine Zahl von 1 bis 9 sein. Die Lösung dieser Rechnung kannst du mit Hilfe des B Knopfes eingeben. Beispiel: Ist die Lösung der Multiplikation 12, dann musst du 12 Mal den B Knopf drücken.

Um das Ergebnis zu überprüfen wird der Microbit geschüttelt. Ist das Ergebnis richtig, wird auf den LED's ein Häkchen angezeigt. Ist das Ergebnis falsch, zeigen die LED's ein X. Im Anschluss wird die richtige Lösung der Rechnung angezeigt.

Was wird gelernt?

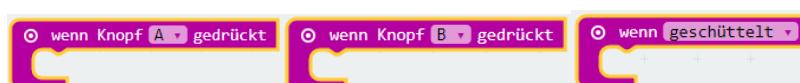
- Generieren von Zufallszahlen
- Verwendung von Variablen
- Anzeige von Symbolen und Zeichenfolgen
- Verwendung der Wenn/Dann/Ansonsten Logik
- Eingabefunktion über die Knöpfe A und B
- Abfrage der Funktion „wenn geschüttelt“

Welche Programmblöcke werden verwendet?

Grundlagen:



Eingabe:



Logik:



Variablen:

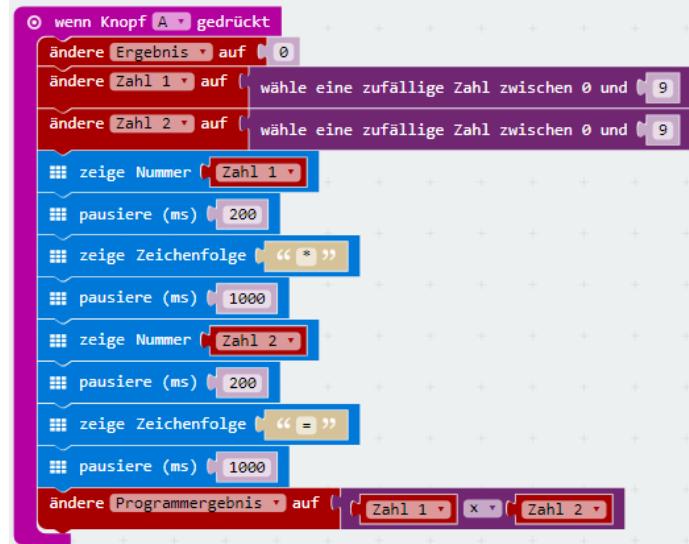


Mathematik



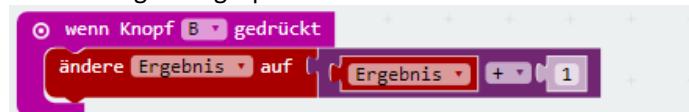
Lösung:

Wenn Knopf A gedrückt wird, wird zuerst die Zählvariable Ergebnis (sie zählt mit, wie oft der Benutzer bei der Eingabe seines Rechenergebnisses den Knopf B gedrückt hat) auf 0 gesetzt, damit wieder richtig mitgezählt werden kann. Im Anschluss werden die Zufallszahlen (Zahl 1 und Zahl 2) generiert und die Rechnung wird angezeigt. Die Rechnung wird vom Programm auch gleich gelöst und in das Ergebnis wird in die Variable Programmergebnis gespeichert.



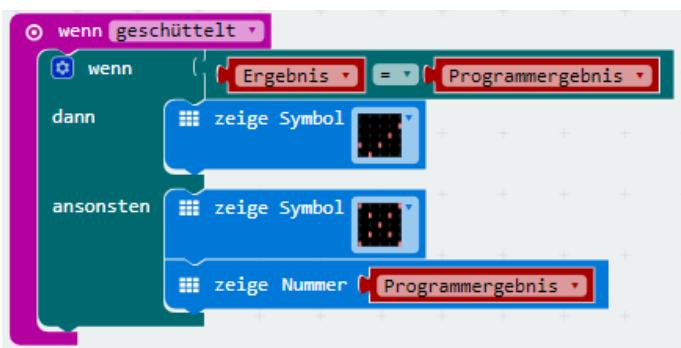
```
when Knopf A gedrückt
  ändere Ergebnis auf 0
  ändere Zahl 1 auf wähle eine zufällige Zahl zwischen 0 und 9
  ändere Zahl 2 auf wähle eine zufällige Zahl zwischen 0 und 9
  zeige Nummer Zahl 1
  pausiere (ms) 200
  zeige Zeichenfolge " * "
  pausiere (ms) 1000
  zeige Nummer Zahl 2
  pausiere (ms) 200
  zeige Zeichenfolge " = "
  pausiere (ms) 1000
  ändere Programmergebnis auf Zahl 1 x Zahl 2
```

In diesem Code-Bereich wird jedes Mal mitgezählt wenn der Knopf B gedrückt wird. Die Anzahl wird in die Variable Ergebnis gespeichert.



```
when Knopf B gedrückt
  ändere Ergebnis auf Ergebnis + 1
```

Hier wird nun noch abgefragt, ob der Microbit geschüttelt wurde. Wenn ja wird die Benutzerlösung Ergebnis überprüft, indem sie mit dem Programmergebnis verglichen wird. Ist das Ergebnis richtig, wird das Symbol Häkchen angezeigt. Wenn nicht, wird das Symbol X und im Anschluss das richtige Ergebnis angezeigt.



```
when geschüttelt
  wenn Ergebnis = Programmergebnis dann
    zeige Symbol Häkchen
  ansonsten
    zeige Symbol Kreuz
    zeige Nummer Programmergebnis
```

8.) Temperaturen messen

Autor: Helmut Lackner

Schwierigkeitsgrad: Mittel

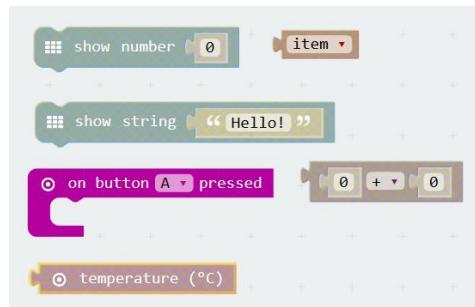
Beschreibung der Aufgabenstellung:

Mit dem eingebauten Temperatursensor des Micro:Bit können Temperaturen gemessen werden. Die Anzeige erfolgt bei Betätigung des Tasters A in Grad Celsius, bei Betätigung des Tasters B in Grad Fahrenheit. Die Umrechnung von C in F erfolgt mit der Formel $F = ^\circ C * 18/10 + 32$ (Dezimalzahlen werden von Micro:Bit nicht unterstützt!) Optional könnte noch bei Betätigung beider Taster die Temperatur mit der Formel $K = ^\circ C + 27315/100$ in Kelvin angezeigt werden

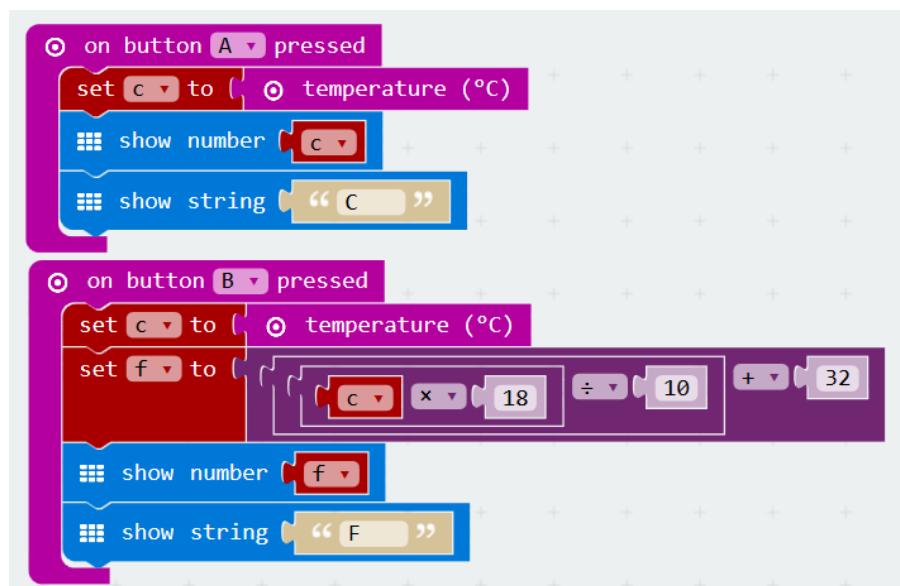
Was wird gelernt?

- Weltweit gibt es unterschiedliche Einheiten, um Temperaturen zu messen.
- Einsatz von Variablen
- Grundrechnungsarten mit Rechenregeln (Punkt vor Strich)

Welche Programmblöcke werden verwendet?



Lösung:



9.) Countdown

Autorin: Gräßl Christina

Schwierigkeitsgrad: Anfänger

Beschreibung der Aufgabenstellung:

Wenn der Micro:Bit geschüttelt wird startet ein Countdown, der von 3 rückwärts zählt. Im Anschluss erscheint ein „GO!“ Signal.

Was wird gelernt?

- Abfrage der „wird geschüttelt“ Funktion
- Ausgabe von Zeichenfolgen bzw. LED's

Welche Programmblöcke werden verwendet?

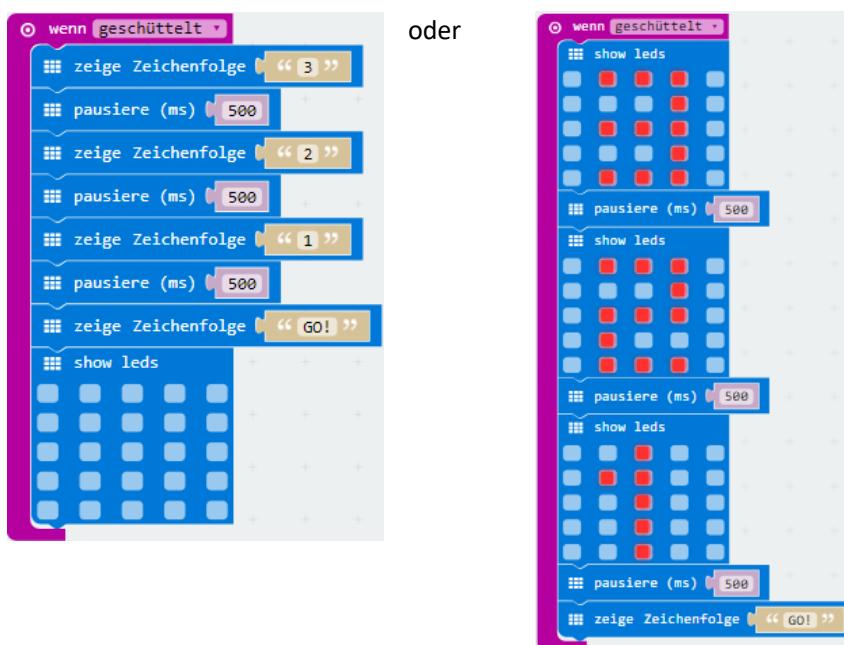
Grundlagen:



Eingabe:



Lösung:



10.) Countdown - Erweiterung:

Autorin: Gräßl Christina

Schwierigkeitsgrad: Mittel

Beschreibung der Aufgabenstellung:

Wenn das Anforderungsniveau gesteigert werden soll, kann man versuchen, dieses Beispiel möglichst kurz zu halten. Hierfür verwendet man dann bereits Variablen und Schleifen.

Was wird gelernt?

- Abfrage der „wird geschüttelt“ Funktion
- Verwendung von Variablen
- Ausgabe von Zeichenfolgen und Variablen
- Verwendung einer Schleife

Welche Programmblöcke werden verwendet?

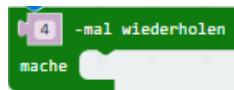
Grundlagen:



Eingabe:



Schleife:



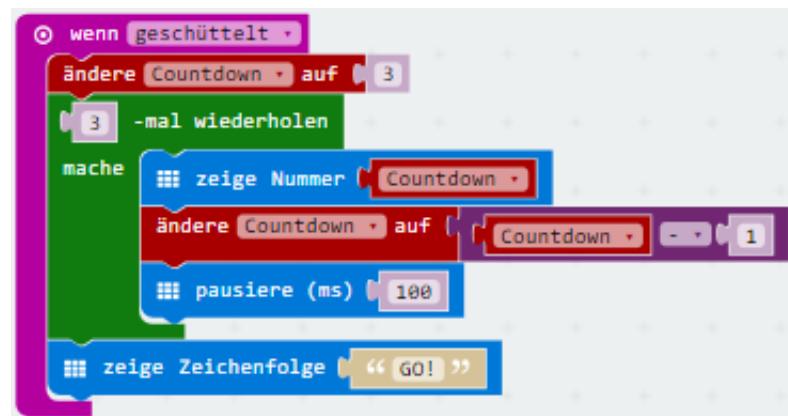
Variablen:



Mathematik:



Lösung:



11.) Schrittzhäler

Name des Autors / der Autorin: Christoph Hobel

Schwierigkeitsgrad: Mittel

Beschreibung der Aufgabenstellung:

Der MICRO:BIT wird mit Hilfe des Programmes zum Schrittzhäler umfunktioniert. Mit der Taste A soll der Schrittzhäler auf die Zahl 0 gesetzt werden. Wird der MICRO:BIT geschüttelt, so soll die Schrittanzahl 2 addiert werden. Mit der Taste B soll die aktuelle Schrittanzahl wieder angezeigt werden.

Was wird gelernt?

- Mit der Taste A sollen die aktuellen Speicherdaten rückgesetzt und neu begonnen werden können.
- Durch das Schütteln soll der MICRO:BIT selbstständig Additionen durchführen.
- Mit der Taste B soll auf Eingabe hin ein Ergebnis abgefragt werden können.

Welche Programmblöcke werden verwendet?



Lösung:



12.) QUIZ Fragen

Autor: Stefan Ouschan

Schwierigkeitsgrad: Fortgeschritten

Beschreibung der Aufgabenstellung:

Man kann mit Hilfe des MICRO:BIT eine wunderbare Überprüfung von analogen Quizfragen generieren. In diesem Beispiel muss man sich im Vorfeld überlegen, wie viel Kapitel möchte man einbauen und wie viele Fragen sollen pro Kapitel erstellt werden. Nach der Erhebung müssen nur mehr einige Parameter verändert werden.

1. Als erstes muss man die einen Überschrift definieren.
2. Danach müssen die möglichen Kapitel des Themenbereiches erstellt werden (wenn möglich nur 1 – 9 Kapitel).
3. Anschließend müssen zu den einzelnen Themenbereichen Fragen erzeugt werden (wenn möglich nur 1 – 9 Fragen). Diese Fragen sollten vier Antworten besitzen, wobei nur ein richtig sein sollte.
4. Die unten angeführte Programmierung erstellt automatisch zu jeder Frage eine passende Lösung, mit den Buchstaben A, B, C und D
5. Der Algorithmus dieses Programmes wird errechnet durch Folgenden Formel

- (Variable **Kapitel** + Variable **Frage**) / 4 = Variable **Rest**
- Variable **Rest** ergibt **Zahlen zwischen 0 bis 3**
- Diese Zahlen werden als Buchstaben dargestellt (1=A, 2=B, 3=C, 0=D)

Dieses Programm lässt sich in allen Gegenständen anwenden und soll den SuS beim Erlernen von Themenbereichen unterstützen.

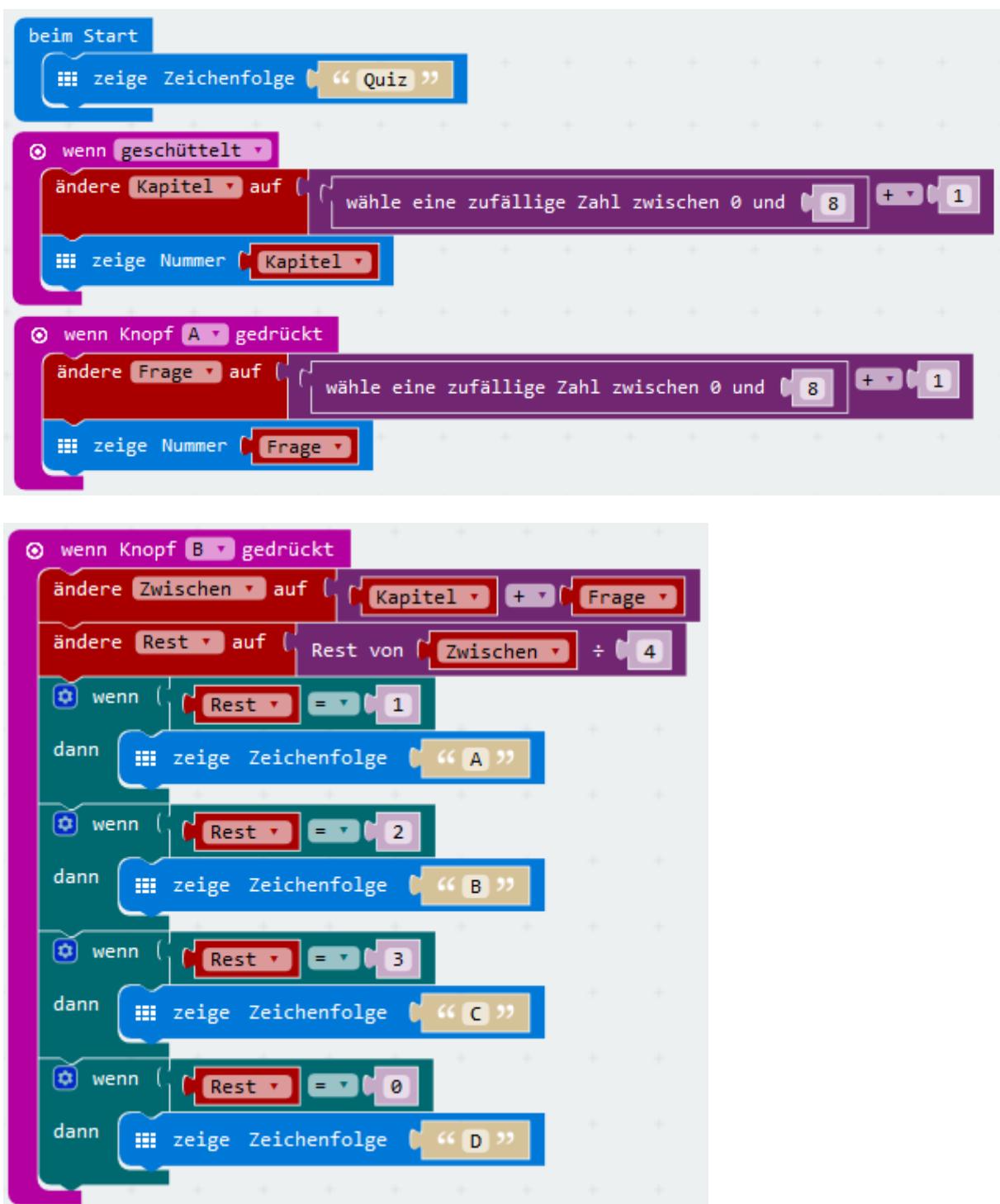
Was wird gelernt?

- Verwenden von einfachen Bausteinen
- Verwenden von verschiedenen Eingabemöglichkeiten
- Verwenden von logische Funktionen (Wenn, dann)
- Verwenden und Begreifen von Variablen
- Verwenden von mathematischen Befehlen

Welche Programmblöcke werden verwendet?

Grundlagen	
Eingabe	
Logik	
Variablen	
Mathematik	

Lösung:



13.) Einfaches Addieren

Autor: Philip Drußnitzer

Schwierigkeitsgrad: Mittel

Beschreibung der Aufgabenstellung:

Erstellen eines einfachen Rechners mit einer Additionsfunktion. Der Micro:Bit kann sich einzelne Variablen merken und damit rechnen. Durch drücken auf die Taste A oder B sollen die einzelnen Zahlen eingegeben werden. Man muss mehrere Variablen (Platzhalter) definieren.

Überlege dir zuerst welche Variablen du brauchen wirst und wie du sie benennen möchtest.

Überlege dir mit welcher Funktion du einzelne Zahlen im Micro:Bit speichern kannst.

Überlege dir mit welcher Funktion du die eingegebenen Daten ausgeben möchtest.

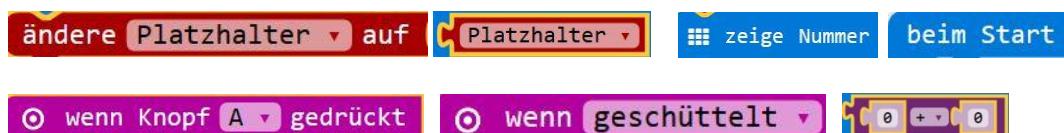
Überlege dir mit welcher Funktion du die Eingabe wieder löschen kannst.

Zum Schluss musst du die letzten Eingaben wieder löschen.

Was wird gelernt?

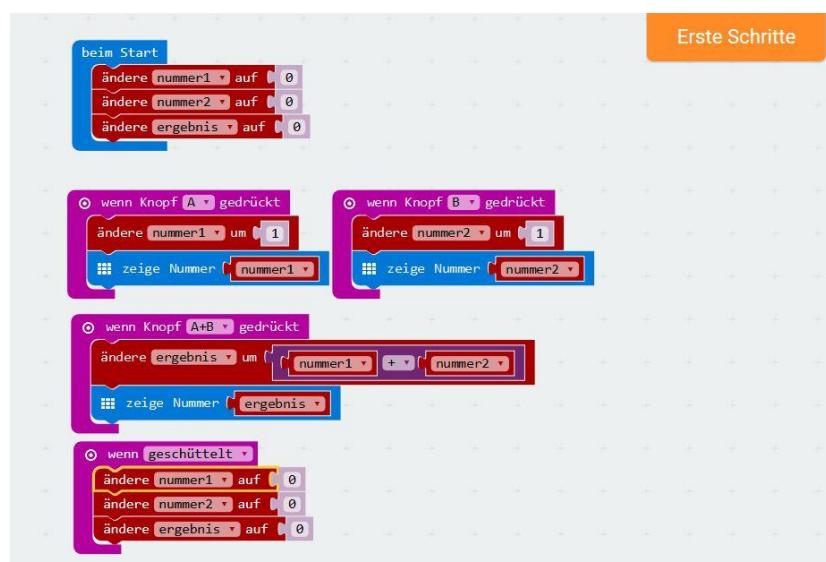
- Definieren von Variablen.
- Definieren von einzelnen Funktionen.
- Konfigurieren von Buttons.
- Konfigurieren des Bewegungssensors.
- Verstehen von Verfahrensabläufen.

Welche Programmblöcke werden verwendet?



Lösung:

Screenshot des Programms (Block):



14.) Schere – Stein - Papier

Name des Autors / der Autorin: Strugl Regina / Harrich Gerhard

Schwierigkeitsgrad: Mittel bis Fortgeschritten

Beschreibung der Aufgabenstellung:

SuS sollen mit dem Micro:Bit das Spiel „Schere – Stein – Papier“ erstellen.

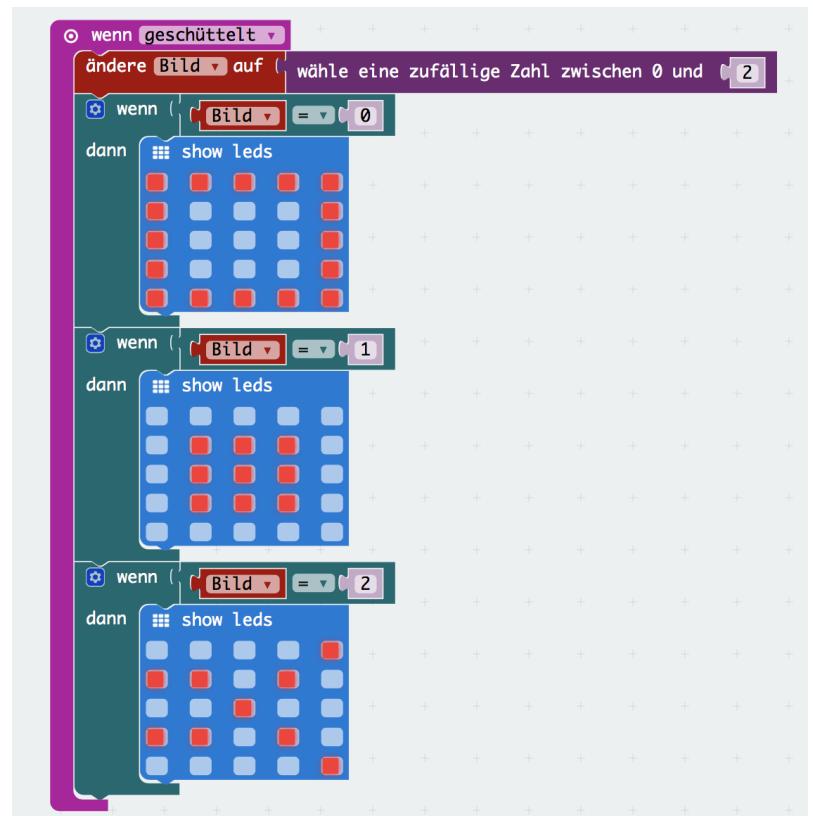
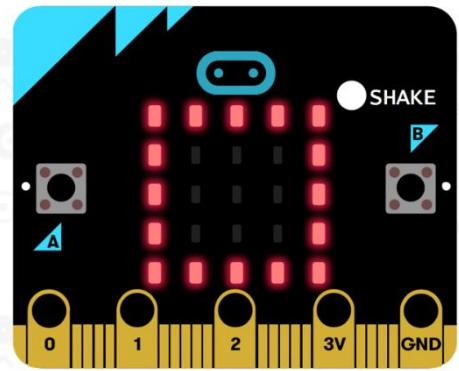
Was wird gelernt?

Nachvollziehen und Ausführen von Handlungsanleitungen (Algorithmen) Einfache Programme mit geeigneten Tools erstellen, um ein bestimmtes Problem zu lösen oder eine bestimmte Aufgabe zu erfüllen.

Welche Programmblöcke werden verwendet?

- Eingabe: wenn geschüttelt
- Grundlagen: show leds
- Schleifen: wenn – dann – Schleife
- Variable: Neu definieren
- Mathematik: Zufallszahl wählen und Variable der Zufallszahl zuweisen.

Lösung:



```
1 let Bild = 0
2 input.onGesture(Gesture.Shake, () => {
3     Bild = Math.random(3)
4     if (Bild == 0) {
5         basic.showLeds(`
6             # # # #
7             # . . . #
8             # . . . #
9             # . . . #
10            # # # #
11        `)
12    }
13    if (Bild == 1) {
14        basic.showLeds(`
15            . . . .
16            . # # # .
17            . # # # .
18            . # # # .
19            . . . .
20        `)
21    }
22    if (Bild == 2) {
23        basic.showLeds(`
24            . . . . #
25            # # . # .
26            . . # . .
27            # # . # .
28            . . . . #
29        `)
30    }
31 })
32
```

15.) Smileys mit dem Micro:Bit anzeigen

Name des Autors / der Autorin: Strugl Regina / Harrich Gerhard

Schwierigkeitsgrad: Anfänger

Beschreibung der Aufgabenstellung:

SuS sollen mit dem Micro:Bit 3 verschiedene Smileys nacheinander anzeigen lassen.

Was wird gelernt?

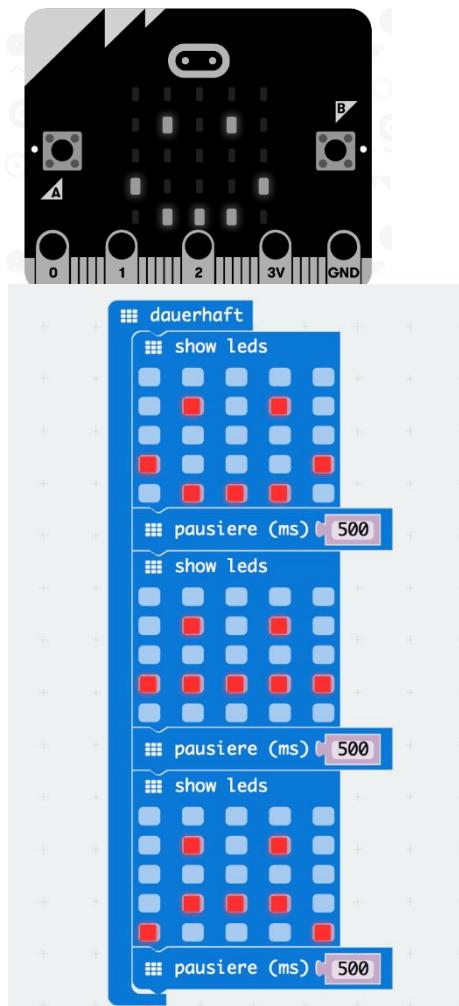
Nachvollziehen und Ausführen von Handlungsanleitungen (Algorithmen). Einfache Programme mit geeigneten Tools erstellen, um ein bestimmtes Problem zu lösen oder eine bestimmte Aufgabe zu erfüllen.

Welche Programmblöcke werden verwendet?

Grundlagen: dauerhaft, show leds (3 x), pausiere (3 x)

Lösung:

Screenshot des Programms (Block)



```
1 basic.forever(() => {
2     basic.showLeds(`
3         . . . .
4         . # . #
5         . . . .
6         # . . .
7         . # # #
8     `)
9     basic.pause(500)
10    basic.showLeds(`
11        . . . .
12        . # . #
13        . . . .
14        # # # #
15        . . . .
16    `)
17    basic.pause(500)
18    basic.showLeds(`
19        . . . .
20        . # . #
21        . . . .
22        . # # #
23        # . . .
24    `)
25    basic.pause(500)
26 }
27 })
```

16.) „Freude schöner Götterfunken“

Name des Autors / der Autorin: Christine Gräßl, Erika Knellwolf, Franz Borotschnig

Schwierigkeitsgrad: Anfänger

Beschreibung der Aufgabenstellung:

Unter Eingabe verschiedener Noten entsteht eine Musiksequenz.



Was wird gelernt?

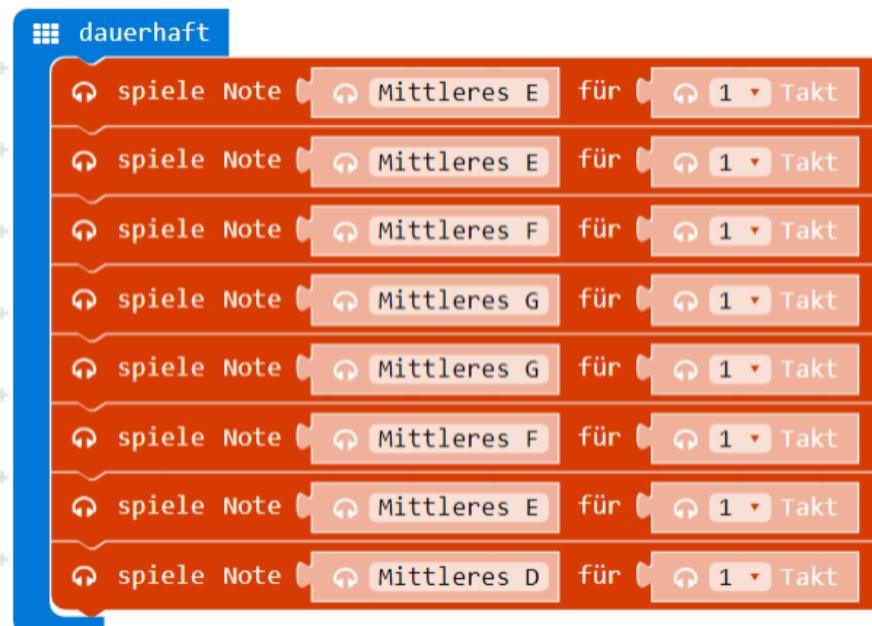
- Programmieren in Einzelschritten
- Anpassen von Parametern in den Programmblöcken

Welche Programmblöcke werden verwendet?



Lösung:

Screenshot des Programms (Block):



17.) Zeig mir die Temperatur

Autor: Andreas Maier

Schwierigkeitsgrad: Anfänger

Beschreibung der Aufgabenstellung:

Wenn der Micro:Bit geschüttelt wird soll er die Raumtemperatur anzeigen

Was wird gelernt?

- Eine Wenn-Verzweigung durch schütteln auslösen.
- Eine neu Variable anlegen.
- Einen in einer Variablen gespeicherten Wert korrigieren können.
- Eine Nummer ausgeben.

Welche Programmblöcke werden verwendet?

① Eingabe	
☰ Variablen	 Neue Variable anlegen ... "Temp"
寁 Mathematik	
grid Grundlagen	

Lösung:

Kommentar:

Der micro:bit besitzt keinen eigentlich keiner Temperatursensor. Es befindet sich jedoch im Prozessor ein Temperatursensor, den man für diesen Zweck verwenden kann. Die Prozessortemperatur liegt üblicherweise etwas über der Raumtemperatur. Die angezeigte Temperatur kann dann durch einrechnen eines Korrekturwerts angepasst werden.

