

The logo of the University of Duisburg-Essen, featuring the text 'UNIVERSITÄT DUISBURG ESSEN' in white capital letters on a dark blue rectangular background.

UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

Interaktive Spieleprogrammierung mit einem Mikrocontroller

Am Beispiel des *BBC micro:bits*

Fatma Yabalioglu

- Medienkompetenzrahmen – Programmierung
- Legitimation der Spieleprogrammierung
- Der *BBC micro:bit*
- Interaktive Spieleprogrammierung – Anwendungsbeispiele
 - „Stein-Schere-Papier“: Blockbasierte Programmierung (MakeCode)
 - „Simon says“: Textbasierte Programmierung (JavaScript)
- Schülerlabor Informatik an der Universität Duisburg-Essen
- Materialpool

Medienkompetenzrahmen – Programmierung

1. BEDIENEN UND ANWENDEN	2. INFORMIEREN UND RECHERCHIEREN	3. KOMMUNIZIEREN UND KOOPERIEREN	4. PRODUZIEREN UND PRÄSENTIEREN	5. ANALYSIEREN UND REFLEKTIEREN	6. PROBLEMLÖSEN UND MODELLIEREN
1.1 Medianausstattung (Hardware) Medianausstattung (Hardware) kennen, auswählen und reflektiert anwenden; mit dieser verantwortungsvoll umgehen	2.1 Informationsrecherche Informationsrecherchen zielgerichtet durchführen und dabei Suchstrategien anwenden	3.1 Kommunikations- und Kooperationsprozesse Kommunikations- und Kooperationsprozesse mit digitalen Werkzeugen zielgerichtet gestalten sowie mediale Produkte und Informationen teilen	4.1 Medienproduktion und Präsentation Medienprodukte adressatengerecht planen, gestalten und präsentieren; Möglichkeiten des Veröffentlichens und Teilens kennen und nutzen	5.1 Medienanalyse Die Vielfalt der Medienentwicklung erkennen, analysieren und beurteilen	6.1 Prinzipien der digitalen Welt
1.2 Digitale Werkzeuge Verschiedene digitale Werkzeuge und deren Funktionsumfang kennen, auswählen sowie diese kreativ, reflektiert und zielgerichtet einsetzen	2.2 Informationsauswertung Themenrelevante Informationen und Daten aus Medienangeboten filtern, strukturieren, umwandeln und aufbereiten	3.2 Kommunikations- und Kooperationsregeln Regeln für digitale Kommunikation und Kooperation kennen, formulieren und einhalten	4.2 Gestaltungsmittel Gestaltungsmittel von Medienprodukten kennen, reflektiert anwenden sowie hinsichtlich ihrer Qualität, Wirkung und Aussageabsicht beurteilen	5.2 Meinungen und Interessen Die Interessen und Verbreitung von Medien erkennen und beurteilen	
1.3 Datenorganisation Informationen und Daten sicher speichern, wiederfinden und von verschiedenen Orten abrufen; Informationen und Daten zusammenfassen, organisieren und strukturiert aufbewahren	2.3 Informationsbewertung Informationen, Daten und ihre Quellen sowie dahinterliegende Strategien und Absichten erkennen und kritisch bewerten	3.3 Kommunikation und Kooperation in der Gesellschaft Kommunikations- und Kooperationsprozesse im Sinne einer aktiven Teilhabe an der Gesellschaft gestalten und reflektieren; ethische Grundsätze sowie kulturell-gesellschaftliche Normen beachten	4.3 Quelldokumentation Standards der Quellenangaben beim Produzieren und Präsentieren von eigenen und fremden Inhalten kennen und anwenden	5.3 Identität und Rollen Chancen und Risiken von Medien nehmen erkennen sowie für die Persönlichkeitsbildung	
1.4 Datenschutz und Informationssicherheit Verantwortungsvoll mit persönlichen und fremden Daten umgehen; Datenschutz, Privatsphäre und Informationssicherheit beachten	2.4 Informationskritik Unangemessene und gefährdende Medieninhalte erkennen und hinsichtlich rechtlicher Grundlagen sowie gesellschaftlicher Normen und Werte einschätzen; Jugend- und Verbraucherschutz kennen und Hilfs- und Unterstützungsstrukturen nutzen	3.4 Cybergewalt und -kriminalität Persönliche, gesellschaftliche und wirtschaftliche Risiken und Auswirkungen von Cybergewalt und -kriminalität erkennen sowie Ansprechpartner und Reaktionsmöglichkeiten kennen und nutzen	4.4 Rechtliche Grundlagen Rechtliche Grundlagen des Persönlichkeits- (u.a. des Bildrechts), Urheber- und Nutzungsrechts (u.a. Lizenzen) überprüfen, bewerten und beachten	5.4 Selbstbestimmung und Medien Medien und deren Nutzung verantwortungsvoll und in der Verantwortung ihrer Medien	

6.3 Modellieren und Programmieren

Probleme formalisiert beschreiben, Problemlösestrategien entwickeln und dazu eine strukturierte, algorithmische Sequenz planen; diese auch durch Programmieren umsetzen und die gefundene Lösungsstrategie beurteilen



Quelle: https://medienkompetenzrahmen.nrw.de/fileadmin/pdf/01_LVR_ZMB_MKR_Rahmen_A4_2018_08_Final.pdf

fatma.yabalioglu@uni-due.de
Interaktive Spieleprogrammierung mit einem Mikrocontroller

Integration der Ziele des Medienkompetenzrahmens NRW (MKR)



Integration der Ziele des Medienkompetenzrahmens NRW (MKR) in die Kernlehrpläne für die Sekundarstufe I des Gymnasiums - Übersicht nach Fächern geordnet -

(Stand: Entwurfss Fassungen der Kernlehrpläne zum 25.2.2019)

Inhaltsverzeichnis

Deutsch:	2
Kunst:	5
Musik:	6
Englisch:.....	7
Französisch:.....	8
Latein:	10
Spanisch:.....	11
Geschichte:	12



Quelle: https://www.schulentwicklung.nrw.de/lehrplaene/upload/klp_SI/Beruecksichtigung_MKR_in_Kernlehrplaenen_GY_SI_2019-03-12.pdf

Integration der Ziele des Medienkompetenzrahmens NRW (MKR)



Ministerium für
Schule und Bildung
des Landes Nordrhein-Westfalen

Suchen

programmieren

Zurück Weiter

Integration der Ziele des Medienkompetenzrahmens NRW (MKR) in die Kernlehrpläne für die Sekundarstufe I des Gymnasiums - Übersicht nach Fächern geordnet -

(Stand: Entwurfss Fassungen der Kernlehrpläne zum 25.2.2019)

Inhaltsverzeichnis

Deutsch: 2

Kunst: 5

Musik: 6

Englisch: 7

Acrobat Reader

Dokument wurde von Adobe Acrobat Reader durchsucht. Keine Treffer.

OK



Quelle: https://www.schulentwicklung.nrw.de/lehrplaene/upload/klp_SI/Beruecksichtigung_MKR_in_Kernlehrplaenen_GY_SI_2019-03-12.pdf

Legitimation der Spieleprogrammierung

Inhaltsfeld 2: Algorithmen (Inhaltliche Schwerpunkte)

- Entwurf von Algorithmen: Die Schülerinnen und Schüler...
 - entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI),
 - beschreiben den Unterschied zwischen der Bottom-Up- und der Top-Down-Methode (A),
 - stellen Algorithmen in verschiedenen Repräsentationen dar (DI),
 - implementieren und kommentieren Algorithmen in einer textorientierten Programmierumgebung (MI),
 - strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI),
 - modifizieren Programme (MI).
- Analyse von Algorithmen: Die Schülerinnen und Schüler...
 - überprüfen Handlungsvorschriften auf Eindeutigkeit und Terminierung (A)
 - beurteilen die Problemangemessenheit eines Algorithmus (A),
 - analysieren und testen Algorithmen und Programme (MI).



Inhaltsfeld 3: Formale Sprachen (Inhaltliche Schwerpunkte)

- Erstellung von Quelltexten: Die Schülerinnen und Schüler...
 - erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache (MI),
 - erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI).
- Analyse von Quelltexten: Die Schülerinnen und Schüler...
 - überprüfen standardisierte Angaben auf formale Korrektheit (A),
 - erläutern die Begriffe Syntax und Semantik an Beispielen (A),
 - analysieren Quelltexte auf syntaktische Korrektheit (A).



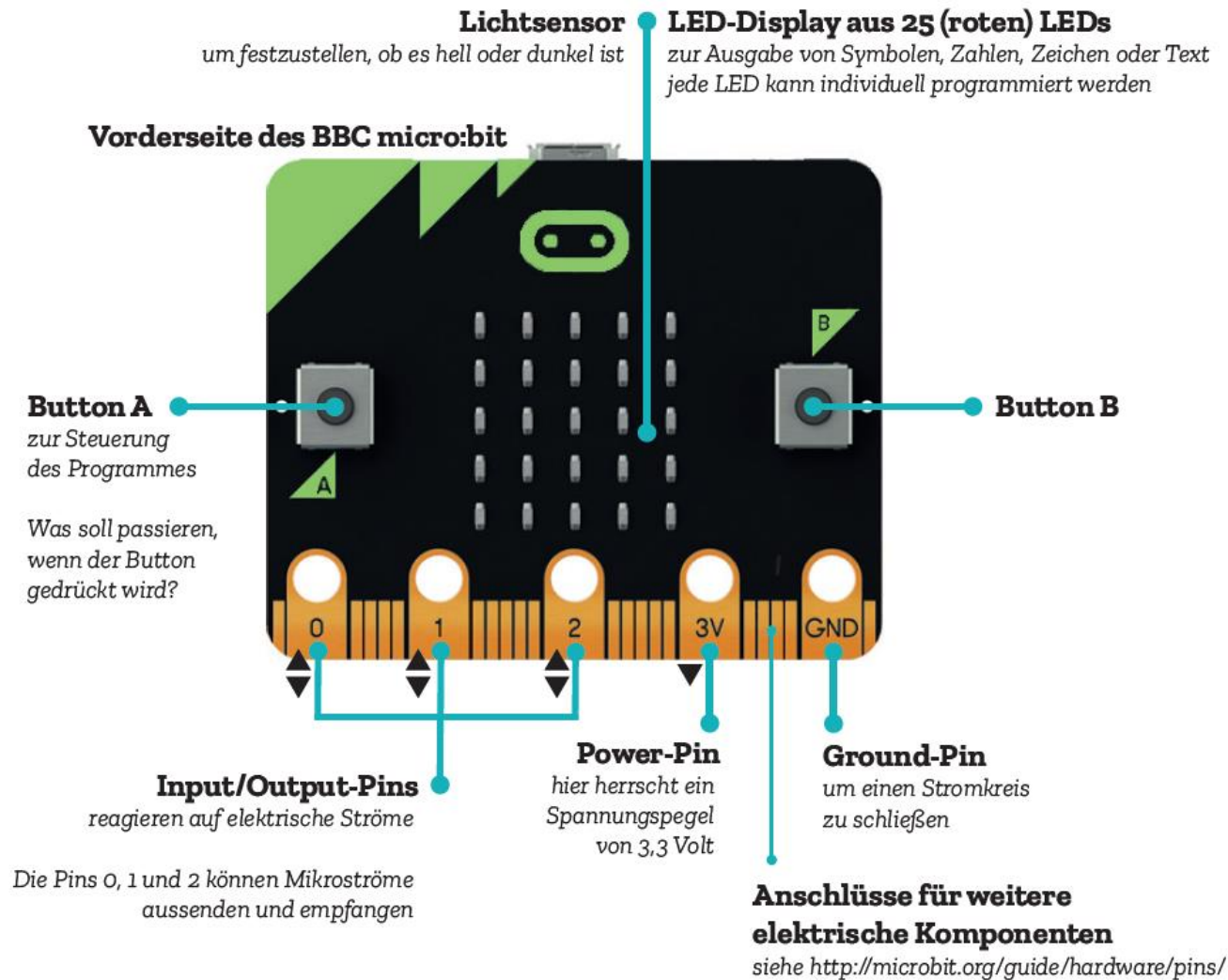
Der BBC micro:bit

Die Funktionen des BBC micro:bit



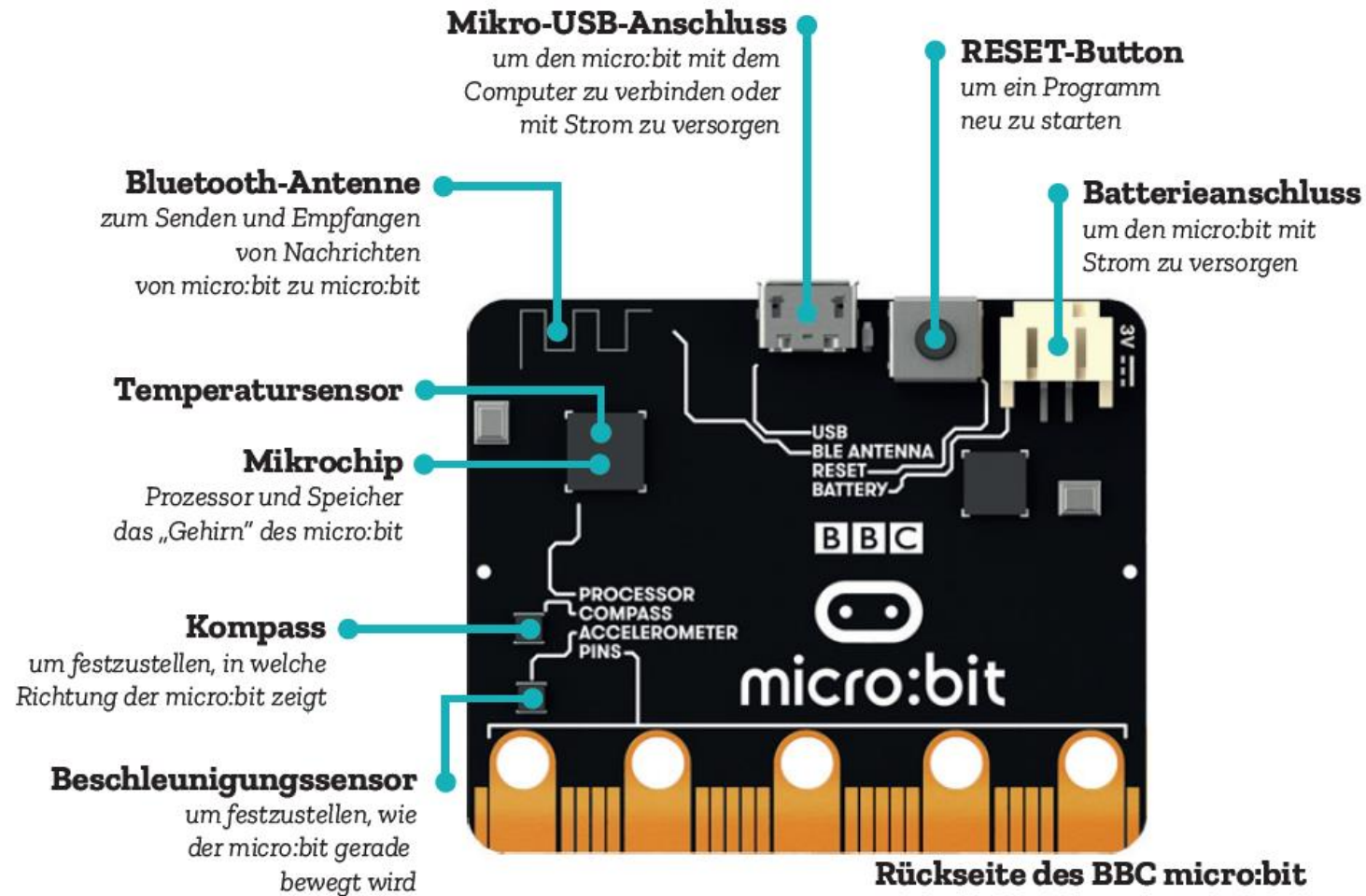
Quelle: https://www.youtube.com/watch?v=zklcA4_GOjc

Die Funktionen des BBC micro:bit – Vorderseite



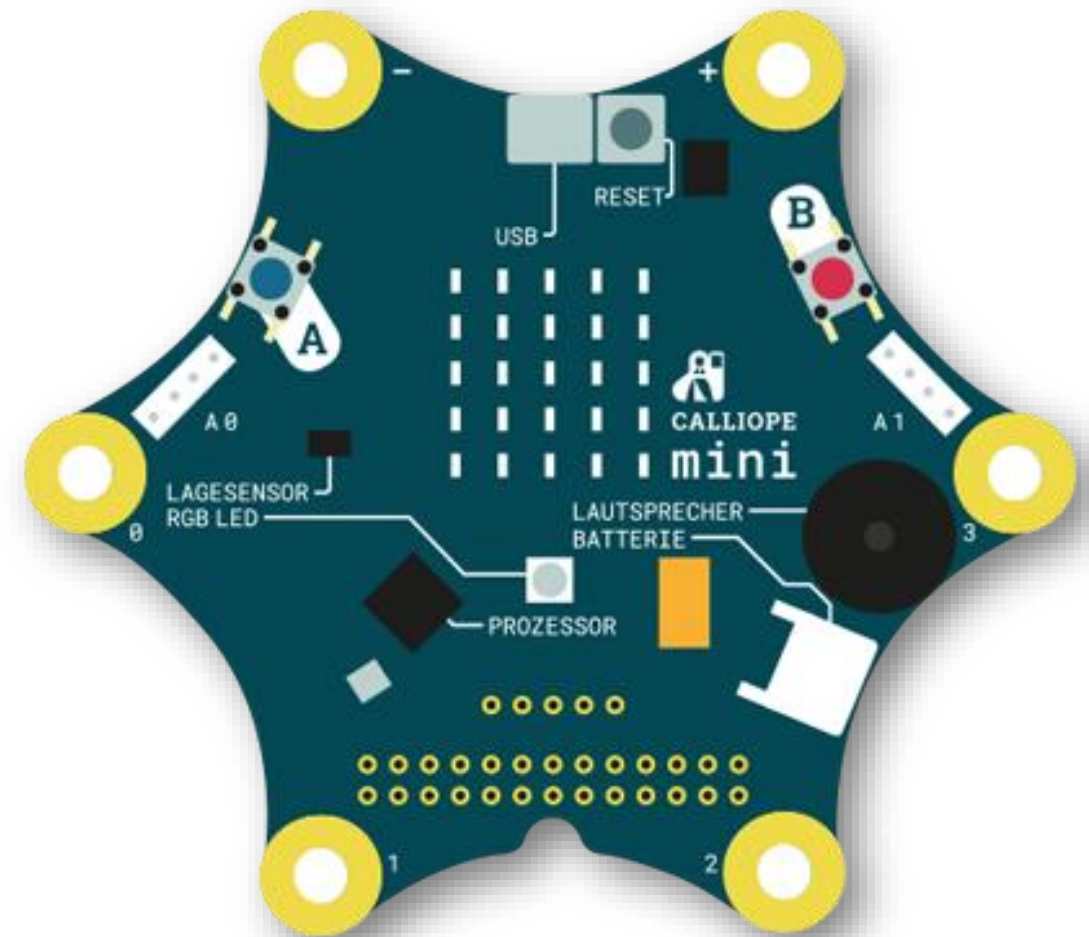
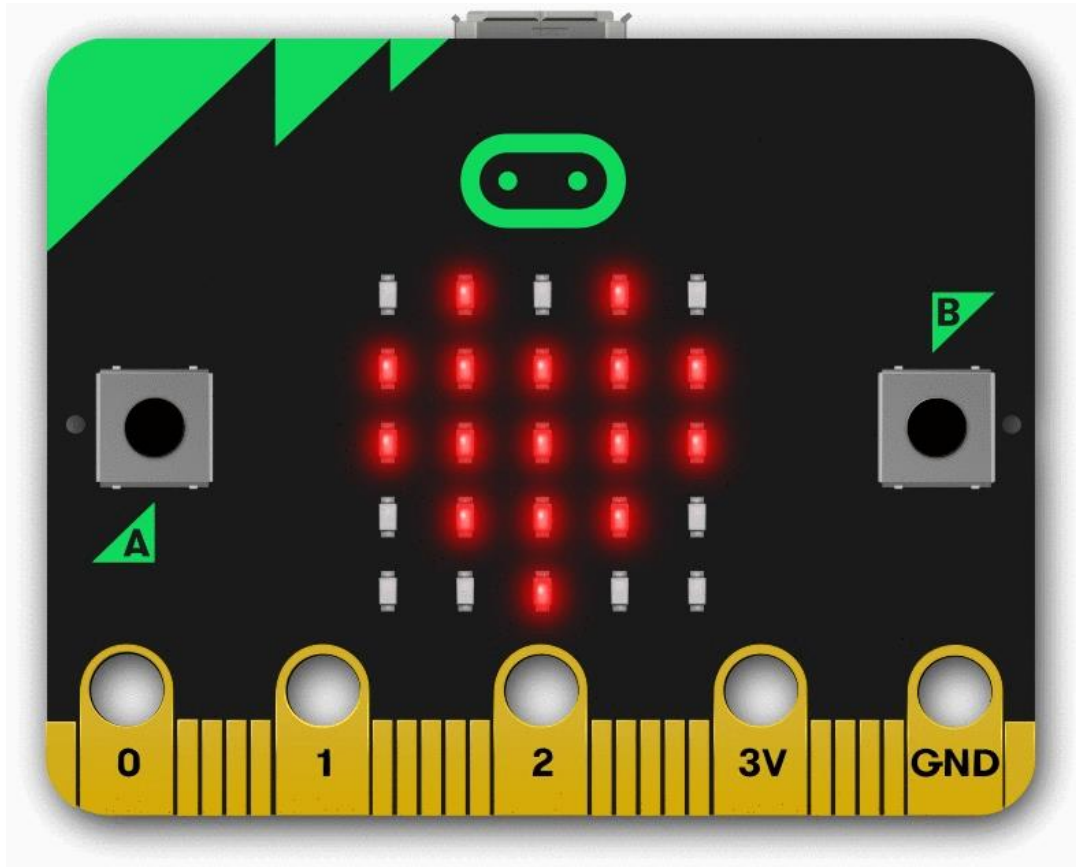
Quelle: Digitale Bildung in der Sekundarstufe – Computational Thinking mit BBC micro:bit (Schulbuch)

Die Funktionen des BBC micro:bit – Rückseite



Quelle: Digitale Bildung in der Sekundarstufe – Computational Thinking mit BBC micro:bit (Schulbuch)

BBC micro:bit vs. Calliope mini



- MakeCode (Microsoft)
 - Online-Editor
 - App für Android, IOS und Windows 10
 - Blockbasiert und textbasiert (JavaScript + Python)

- OpenRoberta
 - Online-Editor
 - Blockbasiert und textbasiert



MakeCode (Microsoft) – Blockbasierte Programmierung

The screenshot displays the MakeCode editor for a micro:bit. The top navigation bar includes the 'micro:bit' logo, 'Startseite', 'Teilen', 'Blöcke', and 'JavaScript' tabs. The main workspace is split into three parts: a visual representation of the micro:bit board on the left, a block palette in the center, and a script area on the right. The block palette lists various categories such as 'Grundlagen', 'Eingabe', 'Musik', 'LED', 'Funk', 'Schleifen', 'Logik', 'Variablen', 'Mathematik', and 'Fortgeschritten'. The script area shows two blue blocks: 'beim Start' and 'dauerhaft'. At the bottom, there is a 'Herunterladen' button, a title field with 'Ohne Titel', and navigation controls.

MakeCode (Microsoft) – JavaScript

The screenshot displays the MakeCode JavaScript editor for a micro:bit. The interface is divided into several sections:

- Top Bar:** Includes the 'micro:bit' logo, navigation links for 'Startseite' and 'Teilen', a 'Blöcke' menu, a selected 'JavaScript' block, and icons for help, settings, and the Microsoft logo.
- Left Panel:** Shows a virtual representation of the micro:bit board with pins labeled 0, 1, 2, 3V, and GND. Below the board are control buttons for running, refreshing, and saving.
- Center Panel:** A search bar labeled 'Suche...' is positioned above a list of block categories: Grundlagen, Eingabe, Musik, LED, Funk, Schleifen, Logik, Variablen, Mathematik, and Fortgeschritten.
- Right Panel:** A code editor showing the following JavaScript code:


```
1 basic.forever(function () {
2
3 })
4
```
- Bottom Panel:** Features a large purple 'Herunterladen' (Download) button, a title field containing 'Ohne Titel', and navigation buttons for undo, redo, and zoom.

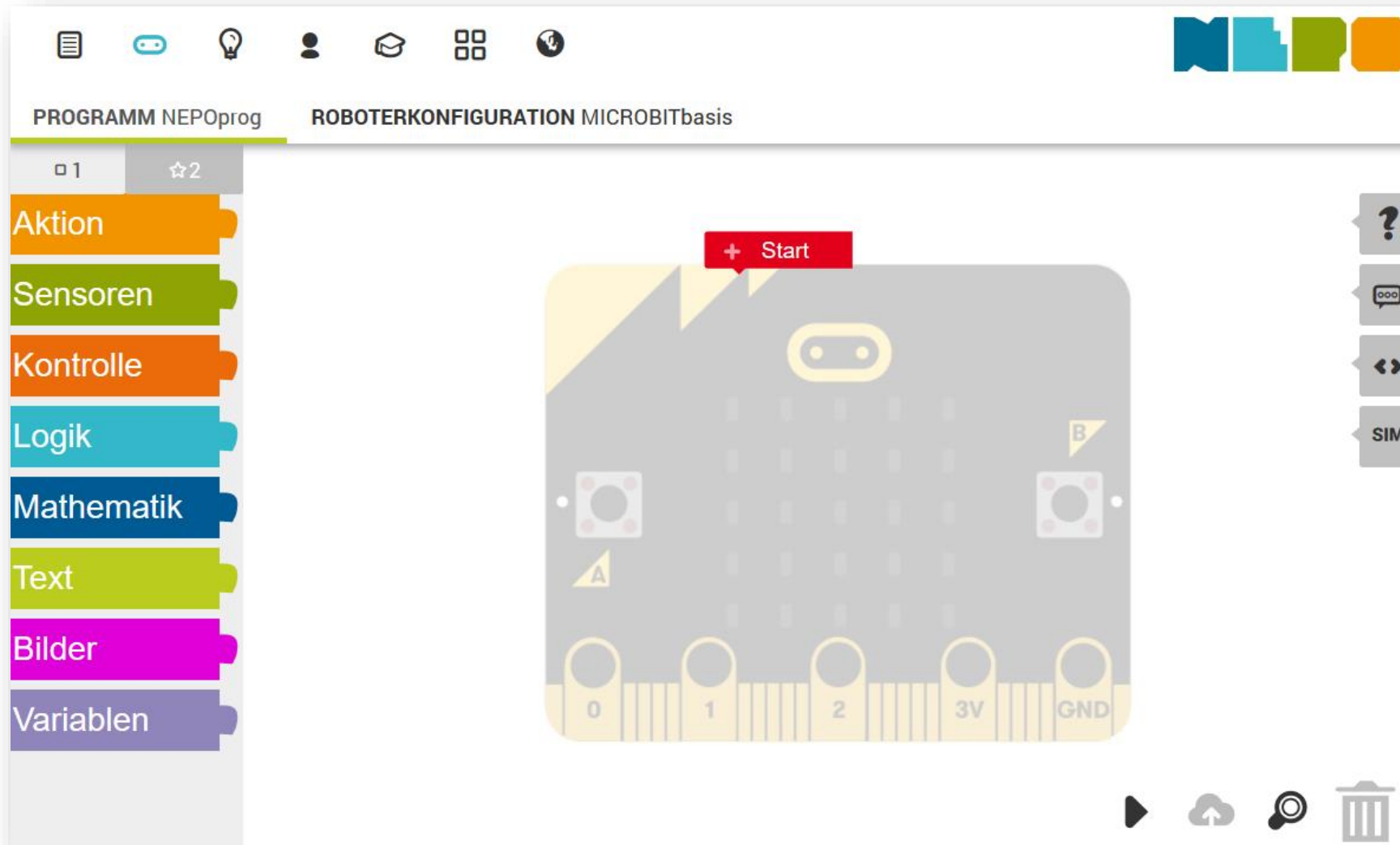
MakeCode (Microsoft) – Python



The screenshot shows the Python Microbit MakeCode editor interface. At the top left, the "micro:bit" logo is displayed. On the right side of the top bar, there is a Python logo. Below the top bar, there is a navigation menu with icons for "Download", "Save", "Load", "Snippets", and "Help". To the right of the menu is a search bar containing the text "microbit". The main area of the editor is a dark background with a light-colored text area containing the following Python code:

```
1 # Add your Python code here. E.g.
2 from microbit import *
3
4
5 while True:
6     display.scroll('Hello, World!')
7     display.show(Image.HEART)
8     sleep(2000)
9
```

OpenRoberta – NEPO: Visuelle Programmierung



OpenRoberta – NEPO: Zusatzfunktionen

The screenshot displays the OpenRoberta NEPO programming environment. The interface is split into two main sections: a visual configuration area on the left and a code editor on the right.

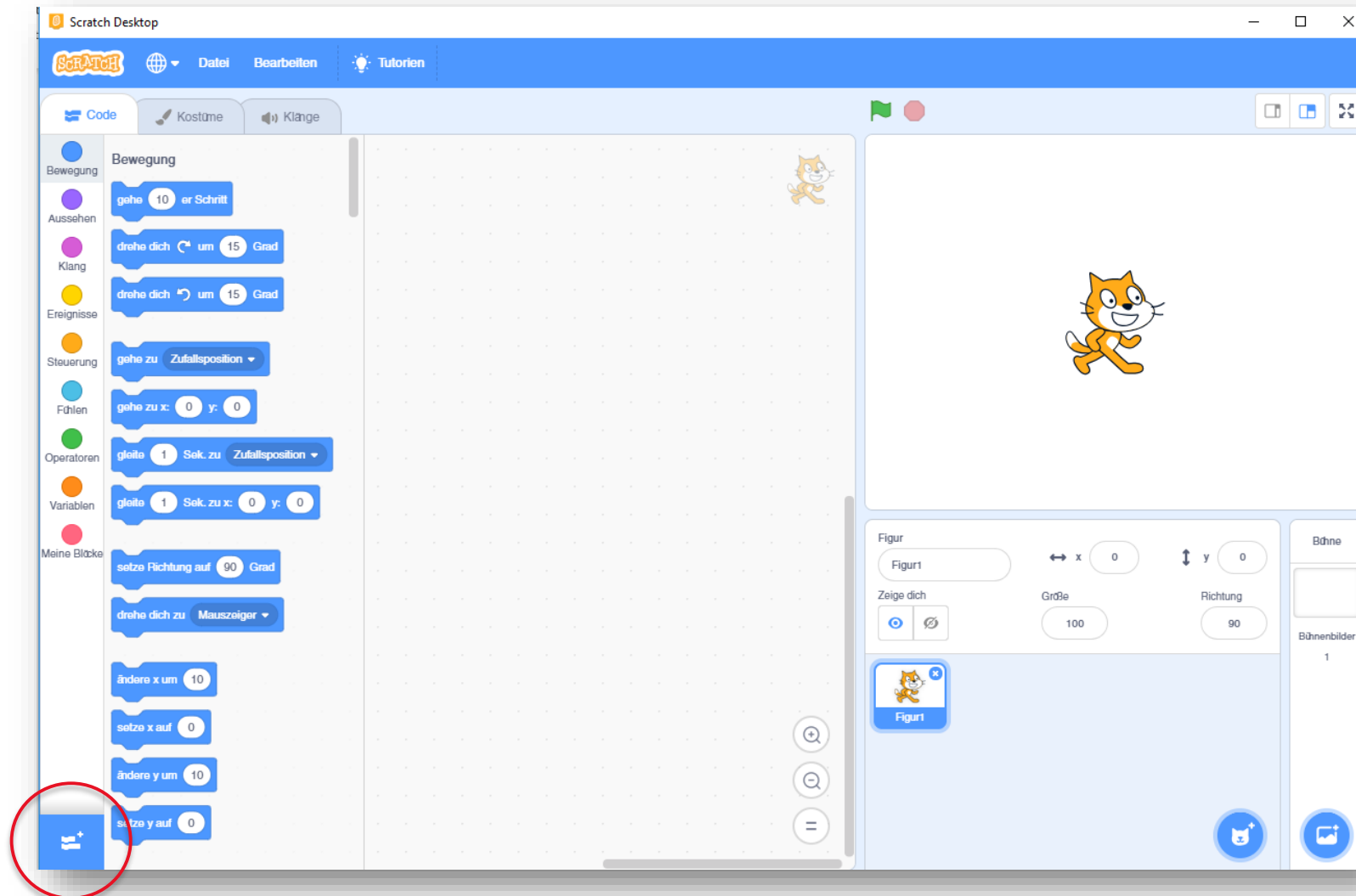
Visual Configuration: The top bar shows icons for a list, chat, lightbulb, user, graduation cap, grid, and refresh. Below this, the tabs are labeled "PROGRAMM NEPOprog" and "ROBOTERKONFIGURATION MICROBITbasis". A sidebar on the left contains icons for various components like a camera, microphone, robot, brain, and text. The main area shows a Micro:bit board with a red "+ Start" button and a double-headed arrow icon. Below the board are icons for play, upload, search, trash, and refresh.

Code Editor: The right side shows Python code for a Micro:bit program. The code includes imports for `microbit`, `random`, and `math`. It defines a constant `_GOLDEN_RATIO` and two custom exception classes: `BreakOutOfALoop` and `ContinueLoop`. The `run()` method uses `microbit.running_time()` and a `global timer1` variable. The `main()` method is wrapped in a `try` block that catches `Exception` and raises it. The code ends with a standard `if __name__ == "__main__":` guard.

```

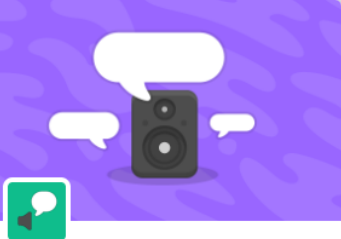



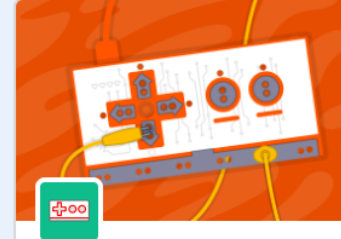
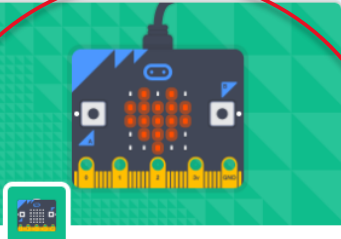



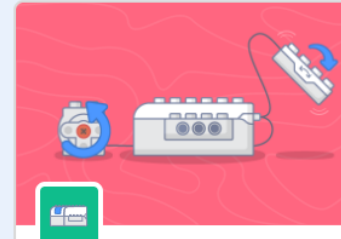

1. import microbit
2. import random
3. import math
4.
5. _GOLDEN_RATIO = (1 + 5 ** 0.5) / 2
6.
7.
8. class BreakOutOfALoop(Exception): pass
9. class ContinueLoop(Exception): pass
10.
11. timer1 = microbit.running_time()
12.
13. def run():
14.     global timer1
15.
16. def main():
17.     try:
18.         run()
19.     except Exception as e:
20.         raise
21.
22. if __name__ == "__main__":
  
```

micro:bit mit Scratch 3.0

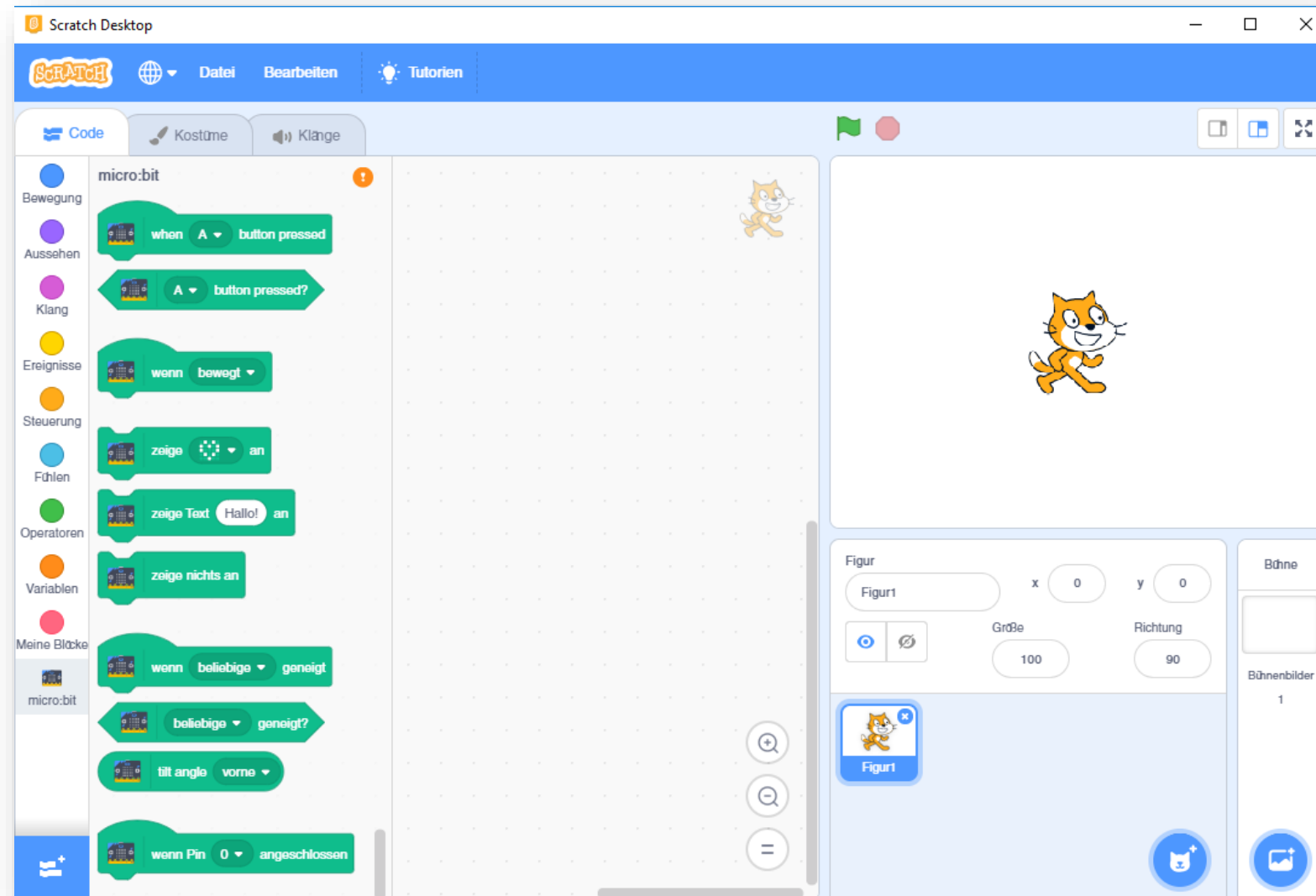


micro:bit mit Scratch 3.0

Scratch Desktop

Musik Spiele Instrumente und Schlagzeug.	Malstift Zeichne mit deinen Figuren.	Video-Erfassung Erfasse Bewegung mit der Kamera.
 Text to Speech Make your projects talk. Requires  Collaboration with Amazon Web Services	 Übersetzen Übersetze den Text in viele Sprachen. Requires  Collaboration with Google	 Makey Makey Make anything into a key. Collaboration with JoyLabz
 micro:bit Verbinde deine Projekte mit der Welt. Requires  Collaboration with micro:bit	 LEGO MINDSTORMS EV3 Baue interaktive Roboter und mehr. Requires  Collaboration with LEGO	 LEGO WeDo 2.0 Baue mit Motoren und Sensoren. Requires  Collaboration with LEGO

micro:bit mit Scratch 3.0



Interaktive Spieleprogrammierung – Anwendungsbeispiele

„Stein-Schere-Papier“: Blockbasierte Programmierung

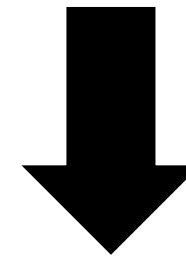
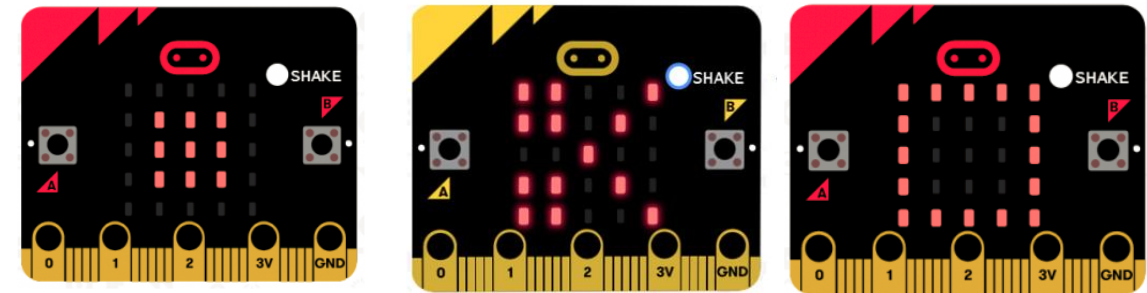


Aufgabenstellung

No-cheat...WAS?

Das BBC micro:bit Wearable kann ein echtes Schmuckstück an deinem Handgelenk sein. Ihr spielt natürlich wie bei der normalen Version des Spiels immer zu zweit. Ausgelöst durch eine starke Schüttelbewegung der Hand wird am micro:bit-Display ein zufälliges Motiv aus den Möglichkeiten SCHERE, STEIN oder PAPIER angezeigt.

Die Regeln kennen wir alle: Schere schlägt Papier, Papier schlägt Stein und Stein schlägt Schere – aber dank des micro:bit völlig schummelfrei!



<https://microbit.eeducation.at/wiki/SchereSteinPapier>

oder

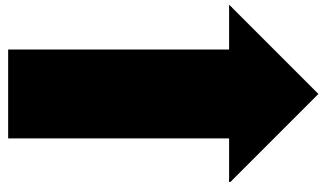
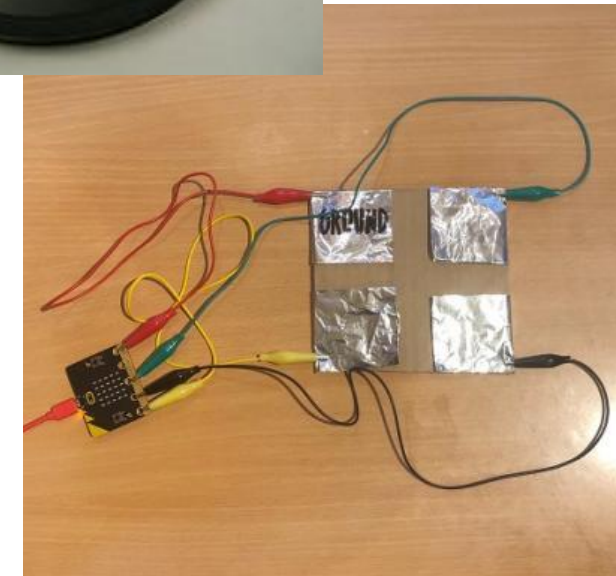
<https://makecode.microbit.org/projects/rock-paper-scissors>

„Simon says“: Textbasierte Programmierung (JavaScript)

Aufgabenstellung

In diesem Workshop wirst du zum Spieleentwickler! Dabei wirst du auf Grundlage des Spieleklassikers "Simon Says", eine Anwendung mit dem BBC micro:bit programmieren. Diese wird dann nicht nur auf dem Rechner funktionieren, sondern auch mit deinem erstellten Spielbrett.

Mit dieser Lerneinheit wirst du erste Erfahrungen in der Programmierung sammeln. Dabei wirst du grundlegende Konzepte der Informatik kennenlernen. Ziel ist es am Ende der Einheit, das selbst erstellte Spiel spielen und testen zu können.



<http://udue.de/simonsays>

„Simon says“: Textbasierte Programmierung (JavaScript)

- https://makecode.microbit.org/_UibgoFcMEKah



Materialpool

Materialpool (siehe Handout)

- Einführungsaufgaben in die Anwendung des micro:bit:
<https://www.schularena.com/ict/informatik/make-it/micro-bit>
- A 14 week Introduction to Computer Science course:
<https://makecode.microbit.org/courses/csintro>
- Informatische Grundbildung in allen Fächern – Werkstattbeispiele zum BBC micro:bit: <https://learninglab.tugraz.at/informatischegrundbildung/bbc-microbit-werkstattbeispiele/>
- Schulbuch: Digitale Bildung in der Sekundarstufe – Computational Thinking mit BBC micro:bit (Österreich)
 - <https://microbit.eeducation.at/index.php/Hauptseite> (Online-Version)
 - https://microbit.eeducation.at/images/f/f2/Buch-microbit_20180729.pdf (Druckversion)
- Zeitschrift zum micro:bit: <https://micromag.cc/>



GESCHICHTE SCHREIBEN (1)

In dieser Einheit sollst du den BBC micro:bit so programmieren, damit dieser später als Hilfsmittel dazu dienen kann, eine Geschichte zu schreiben.

MÖGLICHE ARBEITSSCHRITTE:

1) Das Ziel des Programmes soll es sein, dass durch den BBC micro:bit der Anfangsbuchstabe des nächsten Satzes, den du schreibst, vorgegeben wird. Implementiere also ein Array, welches beim Start des Programmes mit allen Buchstaben des Alphabets gefüllt werden soll. Lass allerdings bei der Implementierung die Buchstaben „C“, „Q“, „X“ und „Y“ weg, da das Geschichtschreiben sonst zu schwierig wird! Solltest du noch nicht mit Arrays vertraut sein, kannst du dazu zuerst im Internet recherchieren!

TIPP: Halte bei der Implementierung des Arrays nach diesem Block Ausschau:



Du wirst diesen Block unter „Fortgeschritten > Arrays“ finden!

Schülerlabor Informatik Universität Duisburg-Essen

Zur Zeit können wir Ihnen folgende Module anbieten:

- Alternate Reality Game: Data Run
- Apps entwickeln mit dem App-Inventor
- Beats & Bits - Programmieren und Musizieren mit Sonic Pi
- Erkundung des Betriebssystems mit BugHunt
- Informatik Enlightened
- Start-up – Netzwerken mit Filius
- Storytelling mit Alice



<http://udue.de/isl>

Mailingliste für Informatiklehrkräfte

<http://udue.de/inflehrer>



Feedback?

Kontakt

Fatma Yabalioglu

Universität Duisburg-Essen

Didaktik der Informatik

Schützenbahn 70, 45127 Essen

fatma.yabalioglu@uni-due.de

<http://udue.de/fy>