

## ARBEITSBLATT

### SIMON

Simon sagt:  
«grün, blau, rot, blau, ...»

### AUFTRAG

Baue ein Spiel, das ähnlich ist wie das Spiel «Simon says».

Programmiere den Calliope so, dass er die Leuchtscheibe in einer zufälligen Folge von Farben leuchten lässt. Danach wiederholen Spielende diese, sie drücken die farbigen Tasten in der gleichen Reihenfolge.

Jede Runde wird die Folge um eine Farbe länger: in der ersten Runde ist nur eine Farbe, dann zwei und so weiter.

Das Spiel ist vorbei, wenn Spielende eine falsche Taste drücken. Dann kann das Spiel neu gestartet werden.

### TIPPS

Es gibt viele mögliche Lösungen.

Die Folge der Farben muss gespeichert werden; das kannst du gut mit einem Array machen.

Nutze die LED-Anzeige vom Calliope: zeige damit an, ob jetzt die Farbfolge als Vorgabe aufleuchtet, ob Spielende nun die Farbfolge wiederholen müssen und so weiter.

Brauche zum Starten des Spiels die Taste A vom Calliope.

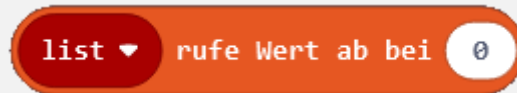
Teile die Aufgabe in Einzelteile (Funktionen) auf und setze sie damit zusammen.

## BLOCKLISTE

Eingabe



Array



### ZUSATZAUFTRAG

Baue eine Hilfestellung ein, welche die nächste Farbe anzeigt, einmal pro Runde.

Lass die Folge der Farben immer schneller leuchten. Dadurch wird das Spiel schwieriger.

Kannst du das Spiel so erweitern, dass es für mehrere Spielende funktioniert?

# ARBEITSBLATT

## SIMON

### BEISPIELLÖSUNG

The code is written in Scratch and is organized into two main sections:

- beim Start (When Started):**
  - setze ledStreifen auf NeoPixels an Pin C17 mit 60 Pixeln und Modus RGB (GRB Format)
  - ledStreifen ausschalten
  - ledStreifen anzeigen
  - setze ledAnzeigeZeit auf 300
  - setze tasteGedrueckt auf falsch
- dauerhaft (Forever Loop):**
  - zeige Text "A"
  - während nicht Knopf A ist geklickt
    - mache pausiere (ms) 100
    - setze spiel auf wahr
    - während spiel
      - mache
        - zeige LEDs (4x4 grid with one lit)
        - Aufruf Simon\_sagt
        - Aufruf Simon\_zeigt
        - zeige LEDs (4x4 grid with four lit)
      - für position von 0 bis Array-Länge abfolge - 1
        - mache
          - wenn spiel dann
            - während nicht tasteGedrueckt
              - mache pausiere (ms) 100
              - setze tasteGedrueckt auf falsch
              - Bildschirminhalt löschen
              - wenn abfolge rufe Wert ab bei position = farbeGeraten dann
                - zeige Symbol für 200 ms
              - ansonsten
                - setze spiel auf falsch
                - zeige Symbol für 200 ms
                - zeige Text "Erreicht"
                - zeige Zahl Array-Länge abfolge - 1 in einem Intervall von 2000 ms
      - pausiere (ms) 1000
      - Bildschirminhalt löschen

```

Funktion setze_farbBuchstabe
wenn farbZahl = 0 dann
  setze farbBuchstabe auf "B"
sonst wenn farbZahl = 1 dann
  setze farbBuchstabe auf "G"
sonst wenn farbZahl = 2 dann
  setze farbBuchstabe auf "R"
ansonsten
  setze farbBuchstabe auf "Y"

```

```

Funktion geraten buchstabe
setze tasteGedrueckt auf wahr
Aufruf setze_farbWahl buchstabe
ledStreifen zeige Farbe farbWahl
pausiere (ms) ledAnzeigeZeit
ledStreifen ausschalten
ledStreifen anzeigen

```

```

Funktion Simon_sagt
setze farbZahl auf wähle eine zufällige Zahl von 0 bis 3
Aufruf setze_farbBuchstabe
abfolge füge Wert farbBuchstabe am Ende hinzu

```

```

Funktion setze_farbWahl buchstabe
wenn buchstabe = "B" dann
  setze farbWahl auf blau
sonst wenn buchstabe = "G" dann
  setze farbWahl auf grün
sonst wenn buchstabe = "R" dann
  setze farbWahl auf rot
ansonsten
  setze farbWahl auf gelb

```

```

Funktion Simon_zeigt
für position von 0 bis Array-Länge abfolge - 1
  mache
    setze farbBuchstabe auf abfolge rufe Wert ab bei position
    Aufruf setze_farbWahl farbBuchstabe
    ledStreifen zeige Farbe farbWahl
    pausiere (ms) ledAnzeigeZeit
    ledStreifen ausschalten
    ledStreifen anzeigen
    pausiere (ms) ledAnzeigeZeit

```

```

wenn Pin P0 gedrückt
  setze farbeGeraten auf "B"
  Aufruf geraten farbeGeraten

```

```

wenn Pin P1 gedrückt
  setze farbeGeraten auf "G"
  Aufruf geraten farbeGeraten

```

```

wenn Knopf A geklickt
  setze abfolge auf leeres Array

```

```

wenn Pin P2 gedrückt
  setze farbeGeraten auf "R"
  Aufruf geraten farbeGeraten

```

```

wenn Pin P3 gedrückt
  setze farbeGeraten auf "Y"
  Aufruf geraten farbeGeraten

```

# ARBEITSBLATT

## SIMON

### RÜCKBLICK

---

Gratuliere zu deinem Spiel! 😊

Du hast gelernt, wie du eine komplizierte Aufgabe lösen kannst.

Weiter hast du gelernt, wie du etwas speichern kannst, das immer grösser wird. Das konntest du mit Arrays machen, die sehr praktisch beim Programmieren sind.

### WIE BIST DU VORGEANGEN?

---

1. Du hast die grosse Aufgabe aufgeteilt in kleinere Teile.
2. Für die Abfrage der Tasten hast du eigene Funktionen erstellt.
3. Du hast den Spielablauf in eine Schleife eingebaut.
4. Du hast ein Array zum Speichern der Farbfolge erstellt und eine Funktion zum Befüllen des Arrays verwendet.
5. Du hast an den richtigen Stellen Pausen und LED-Anzeigen eingebaut.
6. Du hast sich wiederholende Programmblöcke in Funktionen umgeschrieben.

## NEUE BLÖCKE

---

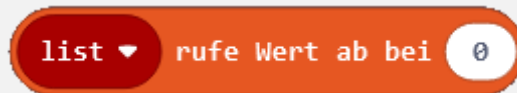
Array



Fügt an das Array «list» am Schluss einen Wert hinzu.



Schaut wie lange das Array «list» ist.



Liest aus dem Array «list» den Wert an der Position 0 aus.