BESUCHER ZÄHLEN MIT CALLIOPE

Calliope ist ein kleiner Minicomputer der mit verschiedenen grafischen Programmiersprachen programmiert werden kann. Wir verwenden in diesem Tutorial die Programmier-"NEPO". Damit sprache kannst du der Calliope ganz einfach vorschreiben, was sie unter welchen Bedingungen zu tun hat. Beispielsweise kannst du bestimmen, dass die Calliope farbige LED an der in einer bestimmten Farbe leuchtet, wenn du eine

Taste drückst. Oder du kannst bestimmen, dass die LED in einer Farbe leuchtet wenn die Taste A gedrückt wird und in einer anderen, wenn die Taste B gedrückt ^{wird.} Oder wen belef wenn die Taste A gedrückt wird und in einer anderen, wenn die Taste B gedrückt ^{wird.} Oder wen belef wenn die Taste A gedrückt wird und in einer anderen, wenn die Taste B gedrückt Du siehst, es gibt verschiedene denkbare Bedingungen und auch verschiedene mögliche Aktionen, mit denen Calliope reagieren kann. Wahrscheinlich mehr als dir so spontan einfallen würden...

In diesem Tutorial wirst du lernen, wie du Calliope eine Anweisung geben kannst und wie du eine Bedingung diese Anwendung definierst. Außerdem wirst für du lernen wie du Calliope dazu bringst, eine Anweisung zu wiederholen, das auch man nennt eine Schleife ausführen. Letztlich wirst du auch lernen eine Variable zu verwenden, deren Wert sich im Laufe des Programms ändern kann.

Am Ende des Tutorials wirst du einige grundlegende Elemente der Programmierung mit NEPO gelernt haben. Das tolle ist, dass praktisch alle Programmiersprachen mit den gleichen Grundelementen arbeiten. Wenn du später also mal mit einer andere Programmiersprache weiterarbeiten willst, nimmst du diese Grundlagen einfach mit.

DAS PROJEKT:

Projekt Dein erscheint zunächst einmal recht überschaubar: Du wirst einen digitalen Besucherzähler (oder dVC für digital Visitors Counter) für Calliope programmieren. Der dVC zählt – zum Beispiel auf Knopfdruck - immer eins zu einem vorhandenen Wert hinzu. Der dVC hat einen älteren Bruder, den analogen Besucherzähler. Vielleicht ist er dir schon einmal begegnet. Sie werden zu verschiedenen Gelegenheiten benutzt, zum Beispiel auf einer Fähre um zu zählen, wie viele Personen tatsächlich an Bord gehen, in Einkaufszentren um zu zählen wie viele Menschen an einem Geschäft vorbeilaufen, bei der Planung von neuen Straßen oder Bahnlinien um zu zählen, wie viele Leute einen Zug nehmen oder eine Straße entlang fahren. Die Einsatzbevielfältig. Fallen sind dir reiche noch andere Anwendungen ein?

Weil das Projekt auf den zweiten Blick doch gar nicht so einfach ist, werden wir es in 5 Schritten entwickeln.

ERSTEN SCHRITT verbindest Ιm du deine Calliope mit Hilfe eines USB-Kabels mit dem Computer und öffnest die Seite <u>https://calliope.cc</u> in einem Browser auf deinem Computer. Dort öffnest du unter dem Punkt "Los qeht's" die Seite

Los geht's Idee Editor Erste Schritte

"Editor". Es stehen verschiedene Editoren zur Verfügung, wir werden in diesem Tutorial mit dem "Open Roberta Lab" arbeiten. Dort musst du noch auswählen, mit welchem Microcontroller-Board du arbeitest. Für



dein Projekt benötigst du die CalliopeMini 2016 oder 2017. Im **ZWEITEN SCHRITT** wirst Du die Calliope dazu bringen, auf einen Tastendruck zu reagieren.



Start

Im Editorfenster gibt es schon den Startblock. An diesen kannst du weitere Blöcke anschließen,

> einfach indem du sie aus dem Menü links in das Editor-Fenster an die entsprechende Stelle ziehst. Das Programm beginnt immer beim Startblock und läuft von dort aus durch die Anweisungen.

Beginne damit eine Anweisung einzufügen: Aktion:

"Schalte LED an Farbe rot" Start Schalte LED an Farbe QLade das Programm auf deine Calliope indem du auf das kleine Play-Symbol klickst und den Anweisungen am Bildschirm folgst. Beobachte, wie Calliope reagiert... Calliope soll aber erst dann reagieren, wenn eine Start Taste gedrückt wurde. Füge also wenn **C** Taste **A** gedrückt? eine Bedingung ein: Kontrolle: mache Schalte LED an Farbe "Wenn Taste A gedrückt dann…"

Überprüfe wieder wie Calliope reagiert.

Das Ergebnis ist sicherlich nicht so wie du es erwartet hast. Das macht aber gar nichts! Tatsächlich funktio-Programm niert das nämlich korrekt. Das Programm startet sofort vom Startblock und prüft, ob die Taste A gedrückt ist. Falls ja, führt es die Anweisung dahinter aus.

Das nennt man iteratives Vorgehen: Eine kleine Veränderung durchführen, die das aktuelle Problem lösen sollte, und prüfen was passiert.

Entweder ist das Problem damit gelöst und wir können uns das nächste Problem vornehmen oder wir haben ein neues Problem erzeugt oder wir haben das ursprüngliche Problem noch nicht gelöst.

Jeden einzelnen Durchlauf durch einen solchen Verbesserungsprozess nennen wir eine Iteration. Es ist ein übliches Vorgehen in der Softwareentwicklung und der Grund, wieso Geduld eine der wichtigsten Tugenden eines Programmierers ist. **FRAGE:** Warum funktioniert das Programm noch nicht so wie gewünscht?

Probiere einmal aus, was passiert, wenn du die Taste A gedrückt hältst, während du Calliope neu startest.



Scanne das Muster mit einem QR-Code-Scanner auf deinem Smartphone für die Antwort!

Damit die LED wie gewünscht auf Tastendruck leuchtet musst du eine + Start Schleife verwenden die Wiederhole unendlich oft die Prüfung immer wieder + wenn C Taste A gedrückt? mache wiederholt. Schalte LED an Farbe mache

Auch jetzt funktioniert das Programm wohl noch nicht wie gewünscht.

FRAGE: Wieso geht die LED nicht aus, wenn du die Taste A loslässt?



Damit die LED auch wieder abgeschaltet wird musst du irgendwo im Programmcode bestimmen, dass das passieren soll. Probiere verschiedene Stellen im Programm aus und teste, wie Calliope reagiert. Wo ist diese Anweisung wohl am sinnvollsten platziert?

Vermutlich hast du beobachtet, dass die LED anfängt zu flackern, wenn du die Taste A drückst. Das liegt daran, dass die Schleife immer wieder durchlaufen wird und zwar sehr schnell hintereinander. Solange die Taste A gedrückt ist, geht die LED pro Durchlauf einmal an und sofort danach wieder aus.

Es gibt verschiedene Möglichkeiten dafür zu sorgen, dass die LED nicht mehr flackert.

nach dem Einschalten Du kannst der LED eine Zeit festlegen für die die LED leuchten soll. Dafür gibt es Warterns **C** 500 bei dem du angeben kannst, den Schalter für welche Zeit (in Millisekunden - also tausendstel Sekunden) das Programm pausieren soll. Du kannst etwas mit der Zeit spielen bis du einen vernünftigen Wert gefunden hast bei dem die LED nicht mehr flackert, bei dem die Taste aber trotzdem noch schnell genug funktioniert.

FRAGE: Was ist bei dieser Lösung problematisch?



Eine bessere Möglichkeit wäre die Verwendung des Schalters

Warte bis gib gedrückt Taste -Α 🗸 = – C falsch –

Dieser etwas kompliziert verschachtelte Schalter sagt dem Programm, es soll warten, bis die Taste A nicht (mehr) gedrückt ist.

Probiere es mal aus!

Spätestens jetzt ist es an der Zeit einige kurze Erklärungen deinen in schreiben, Programmcode um zu erklären, was im einzelnen im Programm passiert.

Dazu qibt es Baustein

👝 Kommentar 📼

den

unter der Rubrik Text. Beschreibe kurz und im folgenden knapp, was Programm-Abschnitt passiert.

WAHRHEITSWERTE

Der Computer arbeitet auf grundlegender Ebene nur mit den Zahlen 0 und 1 weil sich diese sehr einfach und eindeutig durch elektrische Signale darstellen lassen. 0 bedeutet es fließt kein Strom, 1 bedeutet es fließt Strom. Wenn der Computer prüft, ob eine Taste gedrückt ist gibt es auch nur zwei denkbare Antworten: Ja oder Nein. (Wenn die Taste nur so halb gedrückt ist - was gar nicht so einfach ist - dann ist die Taste trotzdem gerade eben noch nicht genug oder gerade eben genug gedrückt und die Antwort ist trotzdem eindeutig Ja oder Nein.) Diese Werte können dann auch gelesen werden als falsch (=0) oder wahr (=1). Wenn der Computer also prüft, ob eine Taste gedrückt ist, gibt er also als Antwort "wahr" oder "falsch" aus bzw. 0 oder 1. In unserem Fall wartet das Programm so lange, bis die Taste A nicht (mehr) gedrückt ist.

Die Erklärung soll knapp aber verständlich sein. Sie hilft dir selbst und anderen dabei das Programm auch nach Wochen, Monaten oder Jahren noch zu verstehen. Einer der häufigsten Anfängerfehler beim Programmieren besteht darin, den eigenen Programmcode nicht ausreichend zu kommentieren. Man macht sich damit das Leben unnötig schwer.

Im **DRITTEN SCHRITT** richtest du eine Variable ein die eine Zahl enthält und sorgst dafür, dass Calliope diese auf Tastendruck erhöht.

Der "dVC" soll ja genutzt werden um Leute zu zählen. farbiq leuchtende Und eine LED hilft uns dabei herzlich wenig. Anstatt eine LED einzuschalten zählen wir zu der Anzahl der Menschen auf Knopfdruck jeweils eins dazu. Dafür benötigen wir erst einmal eine Art "Behälter" in der diese Anzahl gespeichert wird. Wir nennen diesen Behälter eine **Variable**. Du kannst die Variable erzeugen indem du auf das + im Startblock klickst.

+ Start Gib der Variablen gleich einen Namen. Ich habe → Variable Anzahl: Zahl → ← C O den Namen Anzahl gewählt,

du kannst da aber auch etwas kreativer sein. Geschickt ist es den Namen so zu wählen, dass der Name schon verrät, wozu diese Variable im Programm genutzt wird. Das wird Programm sonst später unnötig unübersichtlich. Der Name darf keine Leerzeichen und keine Umlaute oder Sonderzeichen enthalten. Die Variable enthält einfach eine Zahl und wir geben ihr den Startwert 0.

DIE SACHE MIT DEN VARIABLEN

Eine Variable wirkt auf den ersten Blick vielleicht ein bisschen verwirrend oder gar überflüssig. Tatsächlich ist sie aber sehr praktisch. Du kannst dem Programm damit sagen, dass es mit der Zahl "Anzahl der heutigen Gäste" arbeiten soll - und das unabhängig von der tatsächlichen Anzahl der Gäste. Du kannst dir eine Variable also einfach als einen Container vorstellen dem du einen Namen geben kannst und in dem du verschiedene Dinge (z.B. Zahlen, Zeichen oder auch Sätze) ablegen kannst.

Nun müssen wir uns noch darum kümmern, dass beim Drücken der Taste A der Wert der Variablen Anzahl ieweils 1 erhöht kombinieren wir um wird. Dazu Rubriken "Variablen" Programmblöcke aus den und "Mathematik" zu



sonst wenn

mache

Die LED kann dabei gerne weiterhin aufleuchten.

Im **VIERTEN SCHRITT** baust du noch die Möglichkeit ein, den In **Taste B gedrückt?** Mert der Variablen anzuzeigen. Dafür nutzen wir die Taste B und bauen einen weiteren Schritt in die Prüfung ein.

Bei jedem Programmdurchlauf prüft Calliope also erst einmal ob die Taste A gedrückt ist führt ggf. die Anweisungen in dem entsprechenden Block aus und startet einen weiteren Durchlauf der Schleife. Wenn die Taste A nicht gedrückt ist, prüft Calliope ob die Taste B gedrückt ist und führt ggf. die entsprechenden Anweisungen aus bevor sie den nächsten Durchlauf durch die Schleife beginnt.

		ahlia0liah kannat du im fühften scupitt aina Mäaliahkait
+ Start Schileslich kännst du im Tom ill Schkill eine Moglichke.		
Variable Anzahl : Zahl → ← Coo einbauen um den Zahler wieder auf Null zu setzen. Dazu sollen die		
Wiederhole unendlich oft Tasten A und B gleichzeitig gedrückt werden.		
mache	+ -	wenn C Taste A gedrückt? und C Taste B gedrückt?
	mache	beide Tasten exakt gleichzeitig drücken zum zurücksetzen 📼
	sonst we	nn C Taste A gedrückt?
	mache	einen zu Anzahl dazuzählen 📾
		Schalte LED an Farbe C
		+ Warte bis C gib gedrückt Taste A A = C falsch
		Schalte LED aus
	sonst we	enn C Taste B gedrückt?
	mache	Anzahl anzeigen 📾
		Zeige Text - C Anzahl -

FRAGE: Wieso ist es nötig die Prüfung, ob beide Tasten gedrückt sind an den Anfang der Entscheidung zu stellen?



einen funktionierenden Super! Du hast Besucherzähler gebastelt! Jetzt geht es daran ihn ausgiebig in allen möglichen Situationen zu testen. Vielleicht fällt dir noch eine Situation auf in der Calliope nicht so arbeitet wie erwartet, dann müsstest du noch einmal zurück an den Computer und den Fehler beheben.

Nebenbei hast du gelernt, was eine Anweisung ist, wie du verschiedene Anweisungen von Entscheidungen bzw. abhängig machst und wie Bedingungen du einzelne Programmteile mit Hilfe einer Schleife immer wieder durchlaufen kannst. Schließlich hast du auch noch einiges über den Einsatz von Variablen gelernt. Mit diese Grundelementen kannst du schon eine ganze Menge denn alle Programme anstellen basieren auf einer verschachtelten Kombination dieser Grundelemente.

DIESES UNTERRICHTSMATERIAL ist OER. Das bedeutet, dass es für alle frei zugänglich ist und frei genutzt werden kann.



Die Ouellen der verwendeten Bilder und dieses Material selbst ist im Internet zu finden:



https://wiki.zum.de/index.php?title=Besucher_zaehlen_mit_Calliope Erstellt wurde das Material von Konstantin Kowalski 2018.