

Einstieg in das algorithmisches Problemlösen mit dem Calliope im Pflichtfach Informatik - Didaktische Hinweise

Vorbemerkung

Wenn an einer Schule ausreichend *Calliope mini* (s. [1]) oder vergleichbare Sensor-Aktor-Systeme zur Verfügung stehen, können diese zum Einstieg in das algorithmische Problemlösen genutzt werden. Da eine Vielzahl der Informatiksysteme, die uns umgeben, Sensor-Aktor-Systeme sind, werden die Schülerinnen und Schüler damit in ihrer Lebenswelt abgeholt. Es lassen sich eine Reihe von Systemen finden, wie z. B. ein Licht, das abhängig von der Umgebungshelligkeit gesteuert wird, oder ein Temperaturalarm im Kühlschrank, die mit einfachen, überschaubaren Algorithmen rekonstruiert werden können. Dies führt bei den Schülerinnen und Schülern schnell zu einem Kompetenzerleben entsprechende Informatiksysteme nicht nur verstehen, sondern sogar selbst konstruieren zu können. Dabei kann der Grundaufbau des Algorithmus, der z. B. für eine Lichtsteuerung abhängig von der Umgebungshelligkeit benötigt wird, auf weitere Beispiele übertragen werden, indem die verwendeten Sensoren und Schwellenwerte angepasst werden. Die Hürde zur Umsetzung eigener Ideen wird damit möglichst geringgehalten. Durch das Festlegen von Schwellenwerten und das Testen der Systeme erfahren die Schülerinnen und Schüler auch, welchen Einfluss der Programmierer oder die Programmiererin auf die Funktionsweise eines Sensor-Aktor-Systems hat und warum diese Systeme manchmal nicht wie erwartet reagieren.

Wird anschließend mit einem anderen Programmierwerkzeug wie z. B. Scratch (s. [3]) gearbeitet, können die Schülerinnen und Schüler an den Gemeinsamkeiten und Unterschieden der Werkzeuge erkennen, dass algorithmische Kontrollstrukturen wie Schleifen und Verzweigungen keine Besonderheit eines Werkzeugs sind, sondern Grundstrukturen, die unabhängig vom Werkzeug beim algorithmischen Problemlösen eingesetzt werden.

Mit dem Calliope mini steht außerdem ein Werkzeug zur Verfügung, das in anderen Themenbereichen noch einmal eingesetzt werden kann. Beispielsweise um einen eigenen Binärcode zu entwerfen, der es ermöglicht mit den Calliopes Nachrichten über das Funkmodul zu versenden, um das RGB-Modell zu erkunden oder die Hardwarekomponenten eines Informatiksystems exemplarisch zu betrachten.

Steht kein geeignetes Sensor-Aktor-System zur Verfügung kann der Einstieg in die Algorithmik auch mit der blockbasierten Sprache Scratch erfolgen, so dass hier dann etwas mehr Zeit zur Verfügung steht, um bereits vertiefter an Projekten zu arbeiten. Entsprechende Varianten zur Ausgestaltung entsprechender Unterrichtssequenzen sind im Materialpaket zu Scratch enthalten.

Zum Einsatz der Materialien

Die vorliegenden Materialien wurden für die blockbasierte Entwicklungsumgebung Microsoft MakeCode¹ in der Version 7.0.3 erstellt (s. [2]). Die Aufgaben sind für alle drei Versionen des Calliope

¹ Getestet wurden die Materialien mithilfe der MakeCode-Version im Browser (makecode.calliope.cc) und einer USB-Verbindung zum Calliope mini. Bei der Arbeit mit einem Tablet empfiehlt sich die App „Calliope mini“. Beachten Sie ggf. die Hinweise unter <https://calliope.cc/programmieren/mobil/ipad>, um eine Bluetooth-Verbindung zum Calliope mini herzustellen.

mini geeignet. Bei Programmvorlagen ist in MakeCode die passende Version der Hardware auszuwählen.

Die Einheit ist so aufgeteilt, dass zunächst nur die Aktoren des Calliope angesteuert werden, bevor im zweiten Schritt die Reaktion auf Sensoreingaben hinzukommt. Dazu sei angemerkt, dass es in der Programmierumgebung MakeCode ähnlich wie in Scratch die Möglichkeit gibt, auf Ereignisse zu reagieren, die in diesem Fall durch bestimmte Sensorwerte ausgelöst werden. Diese Art der Programmierung bietet für viele Schülerinnen und Schüler einen intuitiven Zugang, so dass diese Möglichkeit häufig bereits beim Erkunden der Aktoren mit einbezogen wird.

Die Materialien sind als Leitfäden angelegt und enthalten daher neben den Aufgaben auch Informationstexte. Diese sind jedoch nicht zwingend zum Selbststudium gedacht. Vielmehr bietet es sich an, die entsprechenden Inhalte gemeinsam im Unterricht zu erarbeiten. Die Aufgaben laden zum Erkunden, Ausprobieren und Hinterfragen ein. Häufig sind sie so angelegt, dass sich die Schülerinnen und Schüler Konzepte an einem Beispiel erarbeiten können. Bei komplexeren Aufgaben können erste Ideen und Lösungsansätze im Plenum diskutiert werden. Die Erfahrungen, die die Schülerinnen und Schüler bei der Bearbeitung der Aufgaben gemacht haben, sollten im Anschluss gesammelt und eingeordnet werden.

Je nach Aufgabe bieten sich für eine Besprechung im Plenum beispielsweise folgende Aspekte an:

- Präsentationen unterschiedlicher Lösungen für das gleiche Problem
- Vielleicht haben einzelne Schülerinnen oder Schüler Aufgaben nach ihren Ideen erweitert und dabei Blöcke oder Konzepte erkundet, die auch für den Rest der Klasse interessant sind
- Austausch über Strategien, wie man vorgehen kann, wenn etwas nicht klappt
- An einem Beispiel Konzepte transparent machen, die sich auf andere Problemstellungen übertragen lassen
- Ein Museumsrundgang, wenn unterschiedliche Lösungen erarbeitet oder unterschiedliche Problemstellungen bearbeitet wurden
- Diskussion von Reflexionsfragen
- Sammlung von Beispielen aus und Bezügen zur Lebenswelt

Die Schülermaterialien liegen als editierbare Dateien vor und sind unter einer CC-BY-NC-SA Lizenz veröffentlicht, so dass beispielsweise auch nur ausgewählte Aufgaben zur Verfügung gestellt oder die Texte reduziert werden können. Die Erklärungen können aber z. B. Schülerinnen und Schülern helfen, die eine Stunde versäumt haben oder später noch einmal etwas nachlesen möchten.

Möglichkeiten der Differenzierung

Der Aufbau der Materialien als Leittexte ermöglicht es leistungsstarken Schülerinnen und Schülern selbständig weiterzuarbeiten und Themen zu vertiefen. Auch einzelne Aufgaben können nach den Ideen der Schülerinnen und Schüler weiter ausgebaut werden.

Die Aufgaben, die sich zur Erarbeitung grundlegender Konzepte in den ersten drei Doppelstunden eignen, sind relativ eng geführt. Je nach Lerngruppe können die Vorgaben reduziert oder bei Bedarf weitere Hilfen zu den einzelnen Aufgaben ergänzt werden. Für leistungsschwächere Schülerinnen und Schüler kann es zu Beginn der Einheit z. B. eine zusätzliche Hilfe sein, eine Vorauswahl an Blöcken vorzugeben. Die Programmierumgebung MakeCode unterstützt diesen Ansatz, indem die

Blöcke, die für die ersten Einstiegsaufgaben benötigt werden, alle in der Kategorie *Grundlagen* zu finden sind.

Das Arbeitsblatt zum Einstieg in die Arbeit mit Sensoren liegt in zwei Versionen vor. Version 2 ist weniger komplex und leitet die Schülerinnen und Schüler stärker an. Der Algorithmus, der hier zur Auswertung des Lichtsensors erarbeitet wird und abhängig vom Schwellenwert zwei verschiedene Reaktionen vorsieht, kann als Grundgerüst für weitere Projekte dienen. Für leistungsschwächere Schülerinnen und Schüler können daher im weiteren Verlauf der Einheit Beispiele ausgewählt werden, auf die dieser Algorithmus übertragbar ist.

Mögliche Unterrichtssequenz

Der folgende Sequenzplan basiert auf Erfahrungswerten in Lerngruppen verschiedener Klassenstufen am Gymnasium. Die vorgeschlagenen Lernziele und Aufgaben sind daher nicht als Minimalziele zu verstehen, sondern müssen je nach Lerngruppe ggf. weiter reduziert werden. Für Lerngruppen die zuvor wenig mit dem schuleigenen System gearbeitet haben, kann es eine besondere Herausforderung sein, die Programme auf den Calliope zu übertragen. Es muss dann ggf. mehr Zeit eingeplant werden, um das Herunterladen und Speichern der Programme zu thematisieren und zu üben. Da die Schülerinnen und Schüler dabei mit verschiedenen Speicherorten in Berührung kommen, ist dies zugleich eine Lernchance, da die Schülerinnen und Schüler hierbei Erfahrungen im Themenbereich „Grundlagen der Datenverarbeitung“ sammeln. Eine Verknüpfung mit diesem Themenbereich bietet sich auch später noch an, indem das EVA-Prinzip und typische Hardwarekomponenten eines Informatiksystems am Beispiel des Calliope thematisiert werden.

Geförderte Kompetenzen

Übergeordnet sollen in der Einheit sukzessive die folgenden Kompetenzen aus dem Modul „technische Realisierung automatisierter Prozesse“ (s. [4]) aufgebaut werden:

Die Schülerinnen und Schüler ...

- lesen Sensoren aus und steuern Aktoren an.
- implementieren einen Algorithmus zur Steuerung einer technischen Komponente.
- *optional: benennen Typen von Sensoren, Aktoren und Verarbeitungskomponenten von technischen Geräten und ordnen sie der Eingabe, Verarbeitung und Ausgabe zu.*

Dabei werden auch bereits Kompetenzen aus dem Modul „Algorithmisieren und Implementieren“ gefördert:

Die Schülerinnen und Schüler

- entwickeln und implementieren einen Algorithmus in einer grafischen Programmiersprache auf experimentelle Weise.
- beschreiben einen Algorithmus in ihren eigenen Worten.
- überprüfen, ob eine Implementierung die Problemstellung löst.

Stunde	Thema der Stunde	Lernziele	Material	Anmerkungen
Die Schülerinnen und Schüler				
1./2.	Erste Schritte mit dem Calliope: einfache Programme zum Ansteuern der Aktoren erstellen	<ul style="list-style-type: none"> können Programme auf dem Calliope speichern und ausführen Anweisungsblöcke zum Ansteuern der Aktoren in die Blöcke „beim Start“ und „dauerhaft“ einfügen den Unterschied der Blöcke „beim Start“ und „dauerhaft“ erläutern die Ausführung von Anweisungen mithilfe des „pausiere“-Blocks verzögern 	01_AB_ErsteSchritte <ul style="list-style-type: none"> Aufgabe 1 und 2 für alle <i>Aufgabe 3 bis 8 dienen der Differenzierung</i> 	Das Erstellen eines Namensschildes soll dazu anregen verschiedene Blöcke zum Ansteuern der Aktoren auszuprobieren. Die Teilaufgaben dienen vor allem unsicheren Schülerinnen und Schülern (SuS) als Orientierung. Wählen experimentierfreudige SuS einen eigenen Erkundungsweg sollte dies unterstützt werden. Erkenntnisse sollten in einem abschließenden Unterrichtsgespräch gesammelt und ausgetauscht werden.
3./4.	Der Calliope reagiert abhängig von Sensorwerten Beispiele: Lichtschalter und abhängig von der Umgebungshelligkeit gesteuertes Licht	<ul style="list-style-type: none"> den Ereignis-Block „Wenn Knopf gedrückt“ verwenden den Wert eines Sensors erkunden und geeignet interpretieren eine Bedingung für einen Schwellenwert eines Sensors erstellen eine Bedingung dauerhaft überprüfen und in einer Verzweigung geeignete Aktionen festlegen 	02_AB1_Sensoren, Aufgabe 1 bis 4 Version1 des AB stellt die zwei Programmieransätze für die Reaktion von Sensorwerten parallel vor und erfordert etwas mehr eigenständiges Problemlösen. Version 2 führt die Programmieransätze	Zum Auswerten der Sensoren stehen zwei Konzepte zur Verfügung: [1] Ereignis-Blöcke, deren Inhalt ausgeführt wird, wenn das entsprechende Ereignis durch bestimmte Sensorwerte ausgelöst wird. [2] Kontinuierlicher Vergleich eines Sensorwerts mit einem Schwellenwert. Dazu wird eine Verzweigung in einer dauerhaft-Schleife ausgeführt. Ziel dieser Doppelstunde ist es, die SuS am Beispiel der Tasten und des Lichtsensors für

			<p>nacheinander ein und leitet die SuS stärker an.</p> <p>Sicherung: 02_AB1a_Sicherung_Grundstruktur</p>	<p>beide Konzepte mit einem Grundgerüst vertraut zu machen. Diese Grundgerüste können sie dann an andere Problemstellungen und Sensorwerte anpassen.</p> <p>Zur Sicherung des Konzepts [2] und Einführung geeigneter Fachbegriffe, kann das AB <i>02_AB1a_Sicherung_Grundstruktur</i> verwendet werden.</p> <p>Voraussetzung für das Festlegen eines Sensorwerts ist das Erkunden und Interpretieren der Sensorwerte. Dies wird hier exemplarisch für den Lichtsensor durchgeführt, sollte aber als allgemeine Strategie thematisiert werden.</p>
5./6.	Übertragung der Konzepte zur Reaktion auf Sensorwerte auf weitere Problemstellungen	<ul style="list-style-type: none"> • für die Problemstellung geeignete Sensoren auswählen und die Sensorwerte passend interpretieren • die algorithmischen Konzepte zur Auswertung von Sensorwerten und zum Festlegen entsprechender Reaktionen auf neue Problemstellungen übertragen. • Herausforderungen und Grenzen bei der Interpretation von Sensorwerten benennen. 	<p>02_AB1_Sensoren, Aufgabe 5</p> <p><i>Aufgabe 6 und die Umsetzung eigener kleiner Miniprojekte sowie</i></p> <p><i>02_AB2_Sensoren_Pins können je nach Zeit und Interesse der SuS eingesetzt werden, auch zur Differenzierung</i></p>	<p>Zu Aufgabe 5 bietet sich eine Sammlung geeigneter Sensoren an. SuS können hier z. B. Lichtsensor, Lagesensor oder das Ereignis „wenn geschüttelt“, das durch den Beschleunigungssensor ausgelöst wird, nennen. Einwände der SuS, dass anhand der Sensorwerte nicht erkannt werden kann, ob der Dieb oder Freddy selbst die Federmappe öffnet, sind erwünscht und sollten zu einer Diskussion über die Zuverlässigkeit und Grenzen von Sensor-Aktor-System genutzt werden.</p>

Mögliche Ergänzung

Sollen das EVA-Prinzip und typische Hardwarekomponenten eines Informatiksystems am Beispiel des Calliope thematisiert werden, kann die folgende Doppelstunde direkt im Anschluss an die oben vorgestellte Sequenz oder auch zu einem späteren Zeitpunkt etwa im Rahmen einer Sequenz zum Thema „Grundlagen der Datenverarbeitung“ durchgeführt werden.

Stunde	Thema der Stunde	Lernziele	Material	Anmerkungen
Die Schülerinnen und Schüler				
1./2.	Das EVA-Prinzip am Beispiel des Calliope	<ul style="list-style-type: none"> das EVA-Prinzip sowohl auf der Ebene eines Programms als auch auf Ebene der Hardware erläutern. Gemeinsamkeiten und Unterschiede des Calliope mit anderen Informatiksystemen wie Smartphone oder Desktop-PC nennen. 	03_AB_EVA-Prinzip	<p>Das EVA-Prinzip wird sowohl auf Software- als auch auf Hardwareebene am Beispiel des Calliope eingeführt. Ein Vergleich der Hardwarekomponenten mit anderen Informatiksystemen zeigt, dass diese bei allen Systemen ähnlich sind und sich im Wesentlichen in Leistungsstärke und Ausprägung unterscheiden.</p> <p>Wenn das Funkmodul noch nicht bekannt ist, kann dieses auch später ergänzt werden.</p>

Zum Weiterarbeiten

Die Aufgaben zur Einführung der Konzepte sind relativ eng geführt. Für einzelne Schülerinnen und Schüler mit Vorerfahrungen oder wenn noch ein bis zwei weitere Doppelstunden zur Vertiefung zur Verfügung stehen, bietet sich daher im Anschluss eine kleine Projektphase an, in der die Schülerinnen und Schüler ein erstes selbst gewähltes Projekt eigenständig umsetzen. Je nach Kreativität der Schülerinnen und Schüler kann die Wahl des Projektes dabei ganz frei bzw. im Rahmen eines gemeinsamen Kontextes oder anhand der beiliegenden Projektvorschläge erfolgen. Um Frustration zu vermeiden, bietet es sich an, die Ideen der Schülerinnen und Schüler ggf. zu sammeln, um einzugrenzen, was sich mithilfe des Calliope umsetzen lässt.

Bei einigen Projekten werden die Schülerinnen und Schüler vermutlich auf Probleme stoßen, die sich ohne die Möglichkeit sich einen Zustand zu merken, nicht lösen lassen. Eine Möglichkeit wäre daher in Jahrgang 10 noch einmal die Implementierung komplexerer Sensor-Aktor-Systeme, die mit Zuständen in Form von Variablen arbeiten, aufzugreifen.

Ergänzende Materialien zur Konstruktion von Sensor-Aktor-System mit dem Calliope und weitere Erläuterungen finden sich im entsprechenden Materialpaket im Lernfeld „Automatisierte Prozesse“.

Wie in den Vorbemerkungen bereits angesprochen, kann der Calliope auch als Werkzeug zur Erkundung von Themen anderer Lernfelder eingesetzt werden.

Die Erweiterung des Calliope um ein Fahrgestell mit zusätzlichen Sensoren (Ultraschallsensor, auf den Boden gerichtete Helligkeitssensoren) eröffnet eine größere Vielfalt an Systemen, die konstruiert werden können, ohne dass dabei zwingend komplexere Algorithmen bzw. Konzepte benötigt werden als im Rahmen dieses Materialpaketes erarbeitet wurden. Steht der Schule eine entsprechende Ausstattung zur Verfügung, kann diese genutzt werden, um motivierende, vertiefende Projekte umzusetzen. Im Lernfeld „Automatisierte Prozesse“ steht beispielsweise ein Materialpaket zur Simulation von Fahrerassistenzsystemen unter Verwendung des Motion Kit V2 (s. [5]) zur Verfügung.

Literaturverzeichnis

- [1] Calliope gGmbH (2020). *Calliope mini – der kleine Computer für große Ideen!* <https://calliope.cc/> [Datum des Zugriffs: 24.05.2022]
- [2] Microsoft (2018). *Microsoft MakeCode. Version 7.0.3* <https://makecode.calliope.cc> [Datum des Zugriffs: 24.03.2025]
- [3] MIT Lifelong-Kindergarten-Group. Scratch. <https://scratch.mit.edu> [Datum des Zugriffs: 16.02.2021]
- [4] Niedersächsisches Kultusministerium (2014). *Kerncurriculum für die Schulformen des Sekundarbereichs I Schuljahrgänge 5 – 10. Informatik*. Hannover: Unidruck
- [5] Tiny Super Lab (2024). *MotionKit 2*. <https://tinysuperlab.com/> [Datum des Zugriffs: 27.03.2025]

Dieses Werk ist lizenziert unter einer [Creative Commons Namensnennung - Nicht kommerziell - Keine Bearbeitungen 4.0 International Lizenz](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Für die korrekte Ausführbarkeit der beiliegenden Quelltexte wird keine Garantie übernommen. Auch für Folgeschäden, die sich aus der Anwendung der Quelltexte oder durch eventuelle fehlerhafte Angaben ergeben, wird keine Haftung oder juristische Verantwortung übernommen.