

# Modul 1: Sequenzen



Wähle die Lektionen aus, die zu Dir und Deiner Klasse passen. Du entscheidest selbst, wie viel Zeit Du Dir für das Thema Sequenzen nehmen möchtest.

## Moduleinführung



Sachanalyse  
Modulgeschichte

Seite 2

## Lektion 1



Analoger Einstieg:  
Sequenzen kennen

Seite 6

## Lektion 2

Gehe **100** Schritte

Basislevel:  
Esuri und der Apfel

Seite 11

## Lektion 3

Drehe **rechts** um **90** Grad

Vertiefungslevel:  
Esuri und die Birne

Seite 16

## Lektion 4



Code-Detektiv:  
Isure und die Erdbeere

Seite 21

## Modulabschluss



Cubi-Mappe

Seite 25

## Zusatzlektion

Sage **“ ”**

Synchronsprechen

Seite 28

## Anhang



Baustein-Lexikon

Seite 33

Das vorliegende Lehrmaterial von IT4Kids und zugehörige Begleitmaterialien für Schüler:innen stehen, soweit nicht anders angegeben, unter der Creative Commons-Lizenz CC BY-NC-SA 4.0. Weitere Informationen zu der Lizenz findest Du hier: <http://creativecommons.org/licenses/by-nc-sa/4.0/>



# Moduleinführung: Sequenzen

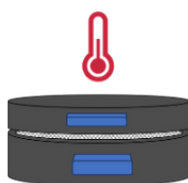
## Sachanalyse

Bevor es richtig mit den Lehrinhalten und dem Programmieren losgeht, möchten wir Dir Wissenswertes rund um die Begriffe **Befehle**, **Algorithmus** und **Programm** mitgeben und erklären, wie sie zusammenhängen.

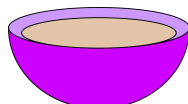
**Was ist ein Befehl?** Wie man bei einem Taschenrechner eine Rechenaufgabe eingibt und sich das Ergebnis ausgeben lässt, kann man auch einem Computer Aufgaben zum Lösen geben. Die Anleitung zum Lösen dieses Problems besteht aus mehreren Schritten.



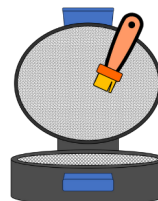
Zur Illustration sei das Beispiel vom Waffelbacken gegeben. Bis die Waffel auf dem Teller liegt, sind verschiedene Arbeitsschritte notwendig:



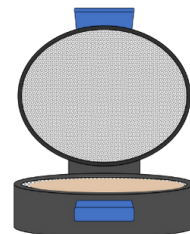
vorheizen



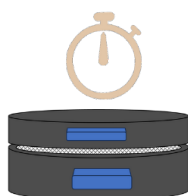
Teig rühren



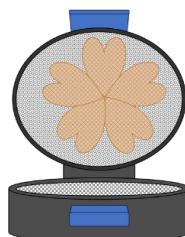
einfetten



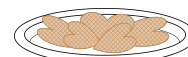
Teig einfüllen



warten



herausnehmen



servieren

Die einzelnen Arbeitsschritte nennt man in der Informatik **Befehle** oder **Anweisungen**. Es ist wichtig zu beachten, dass der Computer nur einen begrenzten Befehlssatz kennt. Befehle, die er nicht kennt, kann er nicht verarbeiten und nicht ausführen.

**Befehle in der Lernsoftware Cubi.** In der Lernsoftware **Cubi** sind die verarbeitbaren Befehle die einzelnen **Bausteine**. Jede Kombination von Bausteinen, die sich zusammenpuzzeln lässt, kann der Computer interpretieren und ausführen.

**Was sind Algorithmen und Programme?** Alle Befehle hintereinander gehängt, die zur Lösung eines Problems beitragen, bilden einen **Algorithmus**. Wird diese abstrakte Abfolge in einer Programmiersprache implementiert, könnte sie ein Roboter ausführen und nicht nur ein Mensch. Diese Implementation nennt man ein **Programm**. Ein Programm ist ein „umgesetzter“ Algorithmus in einer Programmiersprache.

In dem Beispiel mit dem Waffelbacken wäre es wie folgt: Das Rezept und die Erklärung der Arbeitsschritte ergeben zusammen einen Algorithmus zum Waffelbacken. Wenn ich möchte, dass ein Backroboter die Waffeln für mich backt, muss ich den Algorithmus in die Sprache des Backroboters übersetzen. Das Ergebnis nennt man ein Programm.



**! MERKE !**

Befehl: Arbeitsanweisung

Algorithmus: Abstrakte Reihenfolge von Befehlen

Programm: Übersetzung eines Algorithmus in eine Programmiersprache

**Erklärvideo zum Nachschauen.** Alles Wissenswerte rund um Befehle, Algorithmen und Programme kannst Du in dem ersten Erklärvideo unserer Reihe *Was ist ...?* nachschauen. Du findest es hier oder auf unserem YouTube-Kanal.



## Modulinhalte

In diesem Modul steigen die Schüler:innen in die Welt der Informatik ein. Sie lernen, was eine **Sequenz** ist, stellen Überlegungen zu informatischen Inhalten in ihrem Alltag an und programmieren erste Programme in der Lernsoftware **Cubi**.

Das Modul besteht aus vier Lektionen. Die erste Lektion beinhaltet einen **analogen Einstieg** abseits der Tablets, Laptops oder Computer. Hier wird ein Lebensweltbezug für die Schüler:innen hergestellt. Sie überlegen, wo sie in ihrem Alltag Programmierung begegnen und führen Sequenzen händisch durch, indem sie eigene Anleitungen schreiben und ausführen.

Anschließend geht es in der Lektion zum Level **Esuri und der Apfel** ans Programmieren. Dort programmieren sich die Kinder zunächst gegenseitig und navigieren einander durch ein Labyrinth im Klassenzimmer. Anschließend lernen sie die Lernsoftware **Cubi** kennen und entwerfen dort ihr erstes digitales Programm.

Das Vertiefungslevel **Esuri und die Birne** greift die Kompetenzen des Basislevels auf und erweitert sie um den Baustein **Drehe links/rechts um ... Grad**. Um den Schüler:innen die Drehung greifbarer zu machen, wird diese mit einem Bewegungsspiel eingeführt.

Im abschließenden Level **Code-Detektiv: Iru-se und die Erdbeere** begeben sich die Kinder auf Fehlersuche. Ihnen wird ein fertiges Programm vorgelegt, das nicht vollständig durchlaufen kann. Nun liegt es an den Kindern herauszufinden, woran das liegt.

In der Zusatzlektion **Synchronsprechen** können die Schüler:innen ihr Wissen für eine kreative Aufgabe anwenden. Programmierung wird zu einem Werkzeug, mit dem ein Dialog gestaltet wird. Das Theaterstück wird auf der Bühne in Cubi aufgeführt und von den Schüler:innen zeitgleich mündlich vertont. Dadurch üben sich die Kinder nicht nur im Programmieren, sondern auch im Sprachgebrauch und Vortragen.

## Geschichte des Moduls



Raupe Esuri ist gerade aus ihrem Ei geschlüpft und macht ihrem Namen alle Ehre. Denn **Esuri** ist Latein und bedeutet **hungrig sein**. Das ist auch gut so, denn Esuri will möglichst schnell zum Schmetterling werden und dafür muss sie noch wachsen. Doch sie hat ein Problem: Sie findet sich ausgerechnet in einem Ringelblumenbeet wieder! Der intensive Geruch schreckt Esuri ab, sie will die Blumen auf keinen Fall berühren. Zeigt Esuri den Weg zum Essen, so-

dass sie zum Schmetterling werden und aus dem Ringelblumenbeet hinausfliegen kann.

## Die Cubi-Mappe

Die Cubi-Mappe dient als Begleitmaterial für die Schüler:innen. Hier können sie ihre Erkenntnisse sammeln und über ihr Wissen und ihren Lernprozess reflektieren. Das Blatt **Meine Cubi-Mappe** können die Kinder als Deckblatt nutzen. **Der Weg zu Cubi** ist als Merkblatt für die Schüler:innen gedacht. Hier erklärt



Cubi ihnen, wie sie zum Editor kommen. Mit dem Arbeitsblatt **Meine Cubi-Level** können sich die Schüler:innen Levelcodes und ihre bearbeiteten Level mit nach Hause oder in die nächste Unterrichtsstunde nehmen.

Für jedes Modul ist ein Arbeitsblatt hinterlegt, mit dem sich die Schüler:innen ein eigenes **Handout** erstellen können. So kann die Cubi-Mappe während späterer Module oder Unterrichtseinheiten, bei denen die Kinder das Programmieren als Werkzeug verwenden, als Gedankenstütze und Hilfsmittel dienen.

Außerdem gibt es für jedes Modul einen **Selbst-Check**. Der Selbst-Check regt die Schüler:innen zum Reflektieren an. Auf einer ikonischen Skala schätzen die Kinder ihren Wissensstand ein. So wird ihnen bewusst, in welchen Bereichen sie sich sicher fühlen und wo sie noch Herausforderungen sehen.

Die Erweiterung der Cubi-Mappe findet im Modulabschluss statt. Mithilfe der beiden Arbeitsblätter kannst Du Dir einen Überblick über den Lernstand der Kinder verschaffen. Sowohl das Handout als auch der Selbst-Check sind so gestaltet, dass sie bereits nach der Bearbeitung des Basislevels bis auf Knobelaufgaben ausgefüllt werden können. Das gibt Dir die Freiheit, bei den einzelnen Modulen so sehr in die Tiefe zu gehen, wie Du es für sinnvoll hältst.

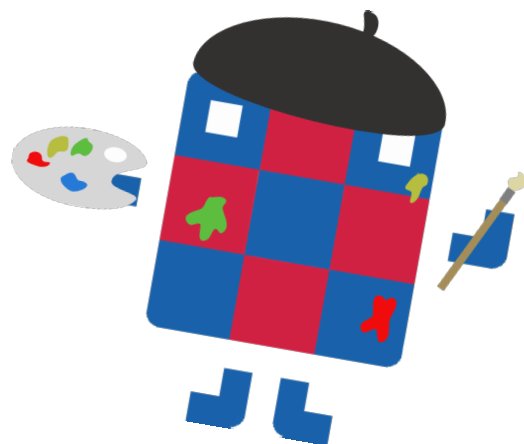
# Analoger Einstieg: Kunstgalerie

## Einführung

In dieser Lektion lernen die Schüler:innen das Konzept und die Relevanz von Programmierung spielerisch kennen. Es werden Ideen und Vorstellungen der Schüler:innen gesammelt, ergänzt und diskutiert. Anschließend stellen sich die Kinder der Herausforderung, präzise Anleitungen zu schreiben, die für ihre Mitschüler:innen nachvollziehbar sind.

## Lernziele

- Die Schüler:innen können das Konzept und Ziel vom Programmieren in eigenen Worten beschreiben.
- Den Schüler:innen sind sich den Herausforderungen bewusst, die durch die Kommunikation mit einem Computer auftreten können.
- Die Schüler:innen wissen, welche Alltagsgegenstände programmierbar sind.



## Tabellarischer Unterrichtsverlaufsplan

| Zeit | Phase                        | Unterrichtsschritte  | SF    | Material   |
|------|------------------------------|--|-------|--|
| 10   | Einstieg                     | Spaziergang durch den Alltag   | T     | Klebezettel, Tafelbild mit Ideen aus dem Alltag                            |
| 5    | Erarbeitung                  | Was bedeutet Programmierung?   | T     | Tafelbild <b>Sequenzen</b>   |
| 5    | Übergang in die Arbeitsphase | Arbeitsauftrag erläutern, Zettel austeilern  | T     | KV <b>Kunstgalerie</b> , AB <b>Meine Sequenz</b> , AB <b>Deine Sequenz</b> |
| 15   | Arbeitsphase                 | Schreiben und Ausführen der Anleitungen  | EA/PA | KV <b>Kunstgalerie</b> , AB <b>Meine Sequenz</b> , AB <b>Deine Sequenz</b> |
| 10   | Präsentation und Reflexion   | Vorstellen der Zeichnungen, Thematisierung von Herausforderungen und Erkenntnissen | P/S   | Zeichnungen und Vorlagen   |

SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit

## Inhalte des Unterrichtsverlaufsplans

### Vorbereitung

Bereite die Unterrichtsstunde vor, indem Du die Wörter aus der Kopiervorlage **Analoger Einstieg Kunstgalerie** zu einzelnen Zetteln schneidest. Du kannst Dir auch eigene Wörter ausdenken! Falte die Zettel, sodass man die Wörter nicht lesen kann. Verstecke die Zettel in einem Hut, einem Sack oder ähnlichem, woraus die Schüler:innen die Zettel ziehen können.

Im Verlauf der Stunde werden mit den Schüler:innen zwei Tafelbilder angelegt. Du kannst zu Beginn der Stunde eine Seite der Tafel zuklappen, sodass nach dem Erstellen des ersten Tafelbilds noch genügend Platz für das zweite Tafelbild bleibt.

## Einstieg

Es wird sich im Tafelkino getroffen. Stell Dir gemeinsam mit den Schüler:innen vor, auf einen Spaziergang durch den Alltag zu gehen. Hier werden Ideen gesammelt, wo den Kindern Geräte begegnen, die programmiert werden. Lass die Schüler:innen die Ideen auf Zetteln notieren und sie an die Tafel hängen. Am besten nutzt Du nur eine Hälfte der Tafel. Die andere kannst Du zunächst zugeklappt lassen. Es entsteht eine Ansammlung, die die Masse und Relevanz von Programmierung im Alltag visualisiert. Beispiele für Programmierung können sein:



- Handy/Smartphone, Laptop, Computer, Tablet
- Fernseher, Radio, Toniebox, CD-/DVD-Player
- Nintendo Switch, PlayStation, XBox
- Elektrische Zahnbürste, Fön, Ventilator, Wasch- und Spülmaschine
- Assistenzsysteme wie Alexa, Siri, Navigationsgeräte
- Elektroautos, Ampeln, Anzeigen in Bus oder Bahn, elektronische Werbetafeln, Parkscheinautomaten

Versuche die Schüler:innen von offensichtlichen Antworten wie Computer, Tablets und Videospielen zu versteckterer Programmierung im Alltag zu leiten und mache so die Allgegenwertigkeit des Themas präsent.

## Erarbeitung

Diskutiere mit den Schüler:innen, was sie sich unter Programmierung vorstellen und was sie bereits über das Thema wissen.

Programmieren bedeutet Anweisungen an einen Computer in der entsprechenden Computersprache zu schreiben. Ein Computerprogramm kann mit einem Kochrezept, einer Bauanleitung für Lego oder mit Musiknoten verglichen werden, denn auch hier wird eine Reihe präziser Anweisungen gegeben, damit das Rezept, die Anleitung oder das Musikstück umgesetzt werden kann. In der Programmierung nennt man eine solche Reihe von Anweisungen eine **Sequenz**.

Notiere die Beispiele für Programmierung ebenfalls an der Tafel. Klappe dafür auch die zweite Hälfte der Tafel auf und nutze die freie Fläche. Schreibe zum Schluss groß **Sequenzen** über die Ideen und sage den Schüler:innen, dass sie in dieser Stunde eine eigene Sequenz in Form einer Anleitung schreiben werden.

## Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen sie eine Sequenz in Form einer Anleitung schreiben. Dazu zieht jedes Kind blind einen der Zettel aus der Kopiervorlage **Analoger Einstieg Kunstgalerie**. Das Zeichen auf dem Zettel ist streng geheim! Es können Zahlen oder Buchstaben sein. Wunder



Dich nicht darüber, dass nicht alle Buchstaben in der Kopiervorlage enthalten sind. Buchstaben, die wie das große **I** nur aus einem Strich bestehen, oder die einem anderen sehr ähneln, wie beispielsweise das kleine **p** dem großen **P**, wurden nicht in die Kopiervorlage integriert. Die Schüler:innen sollen zu dem Zeichen auf ihrem Zettel eine Anleitung auf dem Arbeitsblatt **Meine Sequenz** schreiben. Als Hilfestellung können sie das Zeichen auf dem Zettel nachfahren oder auf einem Schmierpapier testen, in welchen Schritten sie es zeichnen würden. Beim Schreiben der Anleitung müssen sich die Schüler:innen überlegen, welche Informationen wichtig für jemanden sind, der das geheime Zeichen nicht kennt. Anschließend tauschen sie das Arbeitsblatt **Meine Sequenz** mit einem Partnerkind und versuchen dessen Anleitung auf dem Arbeitsblatt **Deine Sequenz** zu befolgen und das Zeichen in den Kasten zu malen.

Ruf den Schüler:innen den Arbeitsauftrag in Erinnerung, indem ein Kind den Auftrag wiederholt.

## Arbeitsphase

Die Schüler:innen arbeiten zunächst in Einzelarbeit. Sie schreiben auf dem Arbeitsblatt **Meine Sequenz** eine Anleitung zu ihrem geheimen Zeichen.

Sobald ein Kind fertig ist, geht es zur Tafel oder einer Haltestelle, die als Treffpunkt im Klassenraum festgelegt wurde, und wartet dort auf ein weiteres Kind. Diese beiden Kinder gehen dann in Partnerarbeit. Alternativ kannst Du die Schüler:innen mit ihren Sitznachbar:innen zusammenarbeiten lassen.

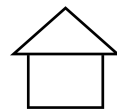


Die Partnerkinder tauschen die Arbeitsblätter **Meine Sequenz**, sodass sie das Blatt des jeweils anderen haben. Nun versuchen sie die Arbeitsanweisungen zu befolgen und auf dem Arbeitsblatt **Deine Sequenz** das geheime Zeichen des anderen zu malen. Wenn beide Partnerkinder fertig sind, vergleichen sie das Zeichen auf dem Zettel mit dem Kunstwerk, das durch die Anweisungen entstanden sind. Konnte das geheime Zeichen herausgefunden werden?



Als Differenzierung nach oben kannst Du schnellen Partnerkindern auftragen zu überlegen, was beim Schreiben und Befolgen der Anweisungen gut geklappt hat und was Herausforderungen waren. Auch kannst du die Partnerkinder neu durchmischen. Das neue Partnerkind versucht dann auf einem separaten Zeichenpapier der Anleitung zu folgen. Bevor die Partnerkinder neu durchgemischt werden, kannst Du ihnen die Gelegenheit geben, ihre Anweisungen zu verbessern

Auch können sich die Kinder an komplexere Formen heranwagen. Die Kopiervorlage **Analoge Einheit Kunstgalerie** enthält neben Zahlen und Buchstaben noch einige Bilder. Zu ihnen ist es schwieriger, eine Anleitung zu schreiben, da sie mehr Bestandteile haben als die Zahlen und Buchstaben. Außerdem kennen die Schüler:innen die Auswahl an Wörtern nicht, sodass sie beim Befolgen der Anleitung sich gegebenenfalls nicht so schnell denken können, worum es sich beim geheimen Zeichen des Partnerkinds handelt.



## Präsentation und Reflexion

Für die Präsentation der Ergebnisse kann eine kleine Ausstellung im Klassenraum veranstaltet werden. Dazu werden jeweils die Originalbilder gemeinsam mit den nachgemalten Bildern ausgelegt. Die Schüler:innen können sich frei im Klassenraum bewegen und die Bilder vergleichen.

Dann treffen sich alle im Sitzkreis und besprechen ihre Erfahrungen und Herausforderungen. Dabei kann auf Schwierigkeiten beim Schreiben der Anleitungen eingegangen werden.

Das Fazit sollte die Erkenntnis sein, dass das Geben genauer Anweisungen eine Herausforderung darstellt. Gleichzeitig brauchen wir genaue Anweisungen, damit das Ergebnis am Ende den Erwartungen entspricht.

## Ausblick

In der nächsten Lektion des Moduls **Basislevel: Esuri und der Apfel** werden sich die Schüler:innen der neuen Herausforderung stellen, ein erstes Level in der Lernsoftware **Cubi** zu programmieren. Dafür wird die Sprache des Computers zunächst mithilfe eines analogen Spiels erarbeitet.



## Basislevel: Esuri und der Apfel

### Einführung

In dieser Lektion lernen die Schüler:innen, eine Sequenz in der Lernsoftware **Cubi** zu programmieren. Dazu wird das Konzept zunächst ganzheitlich und abseits der digitalen Lernumgebung mit dem Spiel **Labyrinth aus Tisch und Stuhl** vermittelt. Anschließend machen sich die Schüler:innen mit der Lernsoftware **Cubi** vertraut und programmieren dort ihr erstes Programm im Level **Esuri und der Apfel**.

### Lernziele

- Die Schüler:innen wissen, was eine Sequenz ist.
- Die Schüler:innen wissen, was ein **Start**- und **Bewegungs**-Baustein ist und in welcher Reihenfolge diese zusammengesetzt werden.
- Die Schüler:innen können Cubi öffnen.
- Die Schüler:innen wissen, wie die Anzahl der Schritte im **Bewegungs**-Baustein geändert werden.

### Geschichte des Levels

Die Raupe Esuri ist gerade aus ihrem Ei geschlüpft und bemerkt das laute Grummeln ihres Magens. Zum Glück liegt nicht weit von ihr ein knackiger Apfel. Kannst Du ihr helfen zum Apfel zu kommen, damit sie groß und stark werden kann?

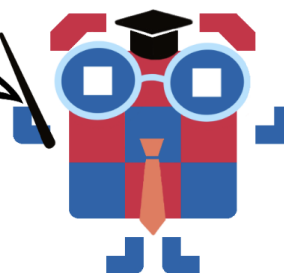
## Tabellarischer Unterrichtsverlaufsplan

| Zeit | Phase                        | Unterrichtsschritte  | SF    | Material                                  |
|------|------------------------------|--|-------|---|
| 5    | Einstieg                     | Rückblick auf den Analogen Einstieg und Wiederholung des Konzepts Programmierung         | P     |   |
| 15   | Erarbeitung                  | Spiel: <b>Labyrinth mit Tisch und Stuhl</b>  | GA    | Tische, Stühle, Rucksäcke, Augenbinde     |
| 10   | Übergang in die Arbeitsphase | Arbeitsauftrag erläutern, Anleitung für den Cubi-Editor, Tablets austeilern              | P     | Tablets/Laptops/PCs, Präsentationstechnik |
| 10   | Arbeitsphase                 | Bearbeitung des Levels <b>Esuri und der Apfel</b>  | EA/PA | Tablets/Laptops/PCs                       |
| 5    | Präsentation und Reflexion   | Vorstellen der Arbeitsergebnisse, Thematisierung von Herausforderungen und Erkenntnissen | T     | Tablets/Laptops/PCs, Präsentationstechnik |

SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit

### Tipp gefällig?

Auf iPads kannst Du Cubi auch über den Startbildschirm starten. Guck dazu gerne in's Cubi-Handbuch unter **Mit Tablets in Cubi** arbeiten.



## Inhalte des Unterrichtsverlaufsplans

### Einstieg

Leite ein Unterrichtsgespräch an, in dem die Schüler:innen ihr Vorwissen aus der analogen Unterrichtsstunde aktivieren. Lass sie sich daran erinnern, wo Programmierung im Alltag zu finden ist und was es mit einer Sequenz auf sich hat. Wenn das Tafelbild aus der vorherigen Stunde noch hängt oder Du Fotos gemacht hast, kannst Du auch nochmal darauf verweisen.

## Erarbeitung

Die Schüler:innen kriegen einen Einblick in die Arbeitsweise eines Programmierers/einer Programmiererin mit dem Computer. Dafür bauen sie zunächst ein Labyrinth aus Tischen, Stühlen und Rucksäcken. Wenn zu wenig Zeit ist, um den Klassenraum umzubauen, können sie auch von ihrem eigenen Sitzplatz zu dem eines anderen Kindes gelotst werden. Die Partnerkinder können den Zielsitzplatz gemeinsam bestimmen. Dann werden zwei Rollen verteilt:

- Das Programmierkind gibt Anweisungen für den Computer. Die Schüler:innen haben sich im Vorfeld abgesprochen wie die Anweisungen aussehen sollen.
- Das andere Kind übernimmt die Rolle des Computers. Dieser kann nicht selbst denken, ihm werden also die Augen verbunden. Er hat keine Kenntnis über seinen Standort und soll lediglich die Anweisungen des Programmierkinds befolgen. Wenn keine Augenbinden vorhanden sind, soll das Computerkind die Augen geschlossen halten. Die Mitschüler:innen kontrollieren, ob das Computerkind die Augen auch tatsächlich zu hat (wer blinzelt, verliert!).

### Arbeitsauftrag:



Die Kinder finden sich in Paaren zusammen und überlegen sich, mit welchen Anweisungen sie das Labyrinth durchlaufen können. Dazu können die folgenden Befehle genutzt werden:

- Gehe einen Schritt.
- Drehe Dich nach links (um 90 Grad).
- Drehe Dich nach rechts (um 90 Grad).

Die Befehle sind bewusst auf ein Minimum reduziert, gerade so, dass das Problem lösbar ist. Dem Computerkind werden dann die Augen verbunden und es wird an den Eingang des Labyrinths gestellt. Das andere Kind gibt nun dem Computerkind Anweisungen, wie dieses innerhalb des Labyrinths ans Ziel gelangen kann.

## Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen sie das Level **Esuri und der Apfel** programmieren, bei dem die Raupe Esuri zu dem Apfel geführt werden muss.

Zeige dafür den Schüler:innen die Lernsoftware **Cubi**. Öffne Cubi über die digitale Tafel oder andere Präsentationstechnik, indem Du im Browser die Seite [editor.i4k.org](http://editor.i4k.org) öffnest. Gehe nun oben links auf **Menü** und anschließend auf **Öffnen** . Es öffnet sich eine Liste an Leveln. Klicke oben im Reiter auf **Module**, um zu den Leveln der Modulreihe zu kommen. Wähle das Level **Esuri und der Apfel** und öffne es. Zeige den Schüler:innen, dass hier programmiert werden kann, indem sie Bausteine aus dem Werkzeugkasten links auf der Programmierfläche in der Mitte zu einem Programm zusammenpuzzeln. Nimm dazu den Startbaustein **Wenn Start gedrückt** und den Bewegungsbaustein **Gehe 100 Schritte**. Wenn sie oben in der Bedienleiste auf den Startknopf  drücken, ist

rechts auf der Bühne zu sehen, was das Programm bewirkt. Mit dem Programm aus dem **Start**- und dem **Bewegungs**-Baustein werdet ihr feststellen, dass Esuri noch nicht bis zum Apfel gelangt. In der Arbeitsphase sollen die Schüler:innen herausfinden, was noch programmiert werden muss, damit die Raupe ihr Ziel erreicht. Bevor sie ihr Programm nochmal starten können, müssen sie ihr Programm zurücksetzen ↺.

Bau nun mit den Schüler:innen gemeinsam die Tablets, Laptops oder Computer auf und stell sicher, dass jedes Gerät mit dem Internet verbunden ist. Lass die Kinder den Cubi-Editor über [editor.i4k.org](http://editor.i4k.org) öffnen und stell sicher, dass alle zu dem Level **Esuri und der Apfel** finden. Zum Öffnen des Editors kann das Merkblatt **Der Weg zu Cubi** als Hilfsmittel dienen. Arbeiten die Schüler:innen an Tablets, können sie den QR-Code einscannen, der sie direkt zu der Vorlage des Levels führt. Eine Kopiervorlage für den QR-Code findest Du im Begleitmaterial.

Ruf den Schüler:innen den Arbeitsauftrag in Erinnerung, indem ein Kind den Auftrag wiederholt.



Levelvorlage:

[level.i4k.org/esuri\\_apfel](http://level.i4k.org/esuri_apfel)

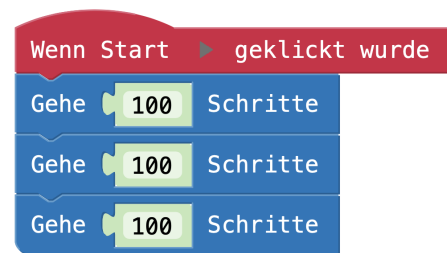


Levellösung:

[level.i4k.org/esuri\\_apfel\\_lsg](http://level.i4k.org/esuri_apfel_lsg)

## Arbeitsphase

In der Arbeitsphase bearbeiten die Schüler:innen das Level **Esuri und der Apfel**. Um das Level zu lösen, brauchen sie den Startbaustein **Wenn Start geklickt**. Anschließend muss Esuri 300 Schritte gehen, um bis zum Apfel zu kommen. Dies gelingt ihr mithilfe des Bewegungsbaustein **Gehe 100 Schritte**. Es gibt mehrere Lösungen. Die Schüler:innen können den Baustein **Gehe 100 Schritte** dreimal aneinanderpuzzeln oder die **100** zu **300** ändern.



Die Arbeitsphase kann in Einzel- oder Partnerarbeit erfolgen. Arbeiten die Schüler:innen zu zweit, ist es empfehlenswert eine Regel einzuführen, bei der die Bausteine abwechselnd von den Partnerkindern in die Programmierfläche gezogen werden. Auf diese Weise wird sichergestellt, dass alle Schüler:innen den Umgang mit **Cubi** üben.

## Präsentation und Reflexion

Für die Präsentation treffen sich alle im Tafelkino. Dort können die Schüler:innen ihre Arbeitsergebnisse vorstellen und darauf eingehen, was gut geklappt hat und was ihnen schwer gefallen ist. Wird mit Tablets gearbeitet, können die Schüler:innen diese mit nach vorne nehmen und zeigen, wie ihre Lösungen aussehen. Ist eine digitale Tafel oder eine ähnliche Präsentationsmöglichkeit vorhanden, können einzelne Lösungswege dort in groß gezeigt werden.

## Ausblick

Um die Inhalte dieses Moduls weiter zu festigen, bietet sich die Lektion **Vertiefungslevel Esuri und die Birne** an. Dort braucht Esuri wieder Hilfe, diesmal um eine Birne zu erreichen. Doch der Weg ist komplizierter als vorher, denn nun muss Esuri auch um Ecken gehen. Deshalb wird hier die Drehung eingeführt. Dies passiert zunächst wieder mittels eines analogen Spiels.

In der **Zusatzlektion: Synchronsprechen** wird das Programmieren einer Sequenz als Werkzeug eingesetzt, um einen kurzen Dialog zu programmieren, welchen sie am Ende der Klasse vortragen. Dadurch üben sich die Kinder nicht nur im Programmieren sondern auch im Sprachgebrauch.

Das Modul **Sequenzen** kann an dieser Stelle abgeschlossen werden. Die Kinder können das Gelernte mit dem Arbeitsblatt **Handout Sequenzen** sichern sowie über ihren Lernstand mithilfe des Arbeitsblatts **Selbst-Check Sequenzen** reflektieren und so ihre Cubi-Mappe erweitern. Mehr dazu findest Du im Kapitel **Modulabschluss**.



## Vertiefungslevel: Esuri und die Birne

### Einführung

Die Schüler:innen haben die Lernsoftware **Cubi** bereits kennengelernt und dort mit Bausteinen der Kategorien **Start** und **Bewegung** eine **Sequenz** programmiert. Nun gilt es, dieses Wissen anhand des leicht komplexeren Levels **Esuri und die Birne** zu vertiefen. Dabei lernen sie einen neuen **Bewegungs**-Baustein kennen: **Drehe links/rechts um ... Grad**. Um den Schüler:innen die Drehung in eine bestimmte Richtung um eine bestimmte Gradzahl näherzubringen, wird diese in einem analogen Spiel eingeführt.

### Lernziele

- Die Schüler:innen haben die Gelegenheit, das Programmieren einer **Sequenz** und die bereits bekannten **Start**- und **Bewegungs**-Bausteine zu wiederholen und zu vertiefen.
- Die Schüler:innen können eine Drehung um 90° beschreiben.
- Die Schüler:innen sind in der Lage, den Baustein **Drehe links/rechts um ... Grad** anzuwenden.

### Geschichte des Levels

Die Raupe Esuri hat schon wieder einen riesen Raupen-Hunger. Hilf ihr durch das Ringelblumenbeet zur Birne. Dort angekommen, ist sie endlich groß und stark genug, um zum Schmetterling zu werden.



## Tabellarischer Unterrichtsverlaufsplan

| Zeit   | Phase                        | Unterrichtsschritte   | SF    | Material                                  |
|--|------------------------------|---|-------|---|
| 5  | Rückblick und Einstieg       | Rückblick auf das Basislevel und Thematisierung von Drehungen | T     |   |
| 15   | Erarbeitung                  | Spiel <b>Winkelwettbewerb</b>                                 | P     | KV <b>Links und Rechts</b> , Tafelbild    |
| 5  | Übergang in die Arbeitsphase | Arbeitsauftrag erläutern, Tablets austeilern und Level öffnen | P     | Tablets/Laptops/PCs                       |
| 15   | Arbeitsphase                 | Bearbeitung des Levels <b>Esuri und die Birne</b>             | EA/PA | Tablets/Laptops/PCs                       |
| 5  | Präsentation und Reflexion   | Vorstellen der Arbeitsergebnisse, Entdeckungen und Probleme   | T     | Tablets/Laptops/PCs, Präsentationstechnik |
| SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit |                              |   |       |   |

## Inhalte des Unterrichtsverlaufsplans

### Rückblick und Einstieg

Es wird sich im Tafelkino getroffen. Dort erinnern sich die Schüler:innen an die vergangenen Unterrichtsstunden und aktivieren ihr Vorwissen. Folgende Fragen können als Gedankenanstöße dienen:

- Was genau ist ein Algorithmus?  
*Eine Anleitung für einen Computer, ein Kochrezept, eine Bauanleitung bei Lego...*
- An welche Bausteine erinnern sich die Schüler:innen?  
*Sie haben bereits die Bausteine **Wenn Start geklickt wurde** und **Gehe ... Schritte** kennengelernt.*

Erinnere die Schüler:innen an den Einstieg des Basislevels **Esuri und der Apfel**. Dort mussten sie sich drehen, um durch das Labyrinth zu kommen. Dazu wurden der Befehl **Drehe Dich nach rechts/links um 90°** verwendet. In dieser Unterrichtsstunde wird es ebenfalls um Drehungen gehen.

## Erarbeitung

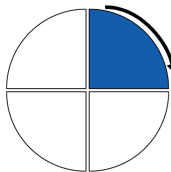
Für die Erarbeitung des Themas **Drehungen** gehen die Schüler:innen zurück an ihre Plätze. Dann startet das Spiel **Winkelwettbewerb**. Dazu stellen sich die Kinder mit dem Gesicht nach vorne zur Tafel hinter ihre Stühle. Es gibt folgende Befehle, die die Kinder auf Zuruf ausführen sollen:

- Mache eine viertel Drehung nach rechts.
- Mache eine viertel Drehung nach links.
- Mache eine halbe Drehung.
- Mache eine ganze Drehung.

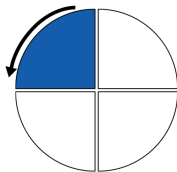
Veranstaltet einen Wettbewerb nach dem Prinzip von **Kommando Pimperle**. Dieses Spiel ist auch unter den Namen **Edgar sagt** oder **Simon Says** bekannt.

- Zu Beginn stehen alle Kinder hinter ihren Stühlen mit dem Gesicht zur Tafel.
- Die Schüler:innen führen die Befehle auf Zuruf aus.
- Die Schüler:innen führen einen Befehl nicht aus, wenn vor dem Befehl kein **Kommando** gerufen wurde, weil der Computer nur die Bausteine ausführt, die mit einem **Start**-Baustein verbunden sind.
- Wenn anstatt **Mache eine ganze Drehung** nur der Befehl **Ganze Drehung** kommt, darf man sich nicht bewegen, weil auch der Computer einen Befehl nur versteht und ausführt, wenn er exakt so formuliert ist wie vereinbart.
- Wenn ein Kind einen Fehler macht, muss es sich hinsetzen. Wer schafft es ins Finale?
- Um die Schwierigkeit zu erhöhen, kannst Du mitmachen und Fehler einbauen.

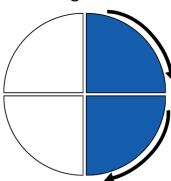
Mache eine Drehung um  $90^\circ$  nach rechts.



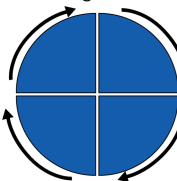
Mache eine Drehung um  $90^\circ$  nach links.



Mache eine Drehung um  $180^\circ$ .



Mache eine Drehung um  $360^\circ$ .

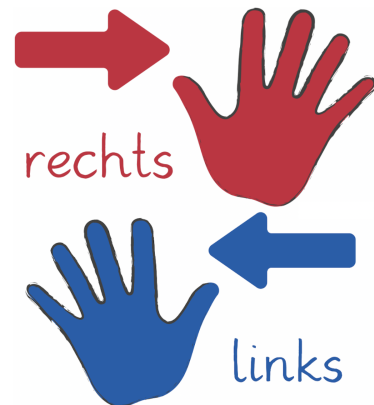


Unterbrich den Winkelwettbewerb kurz, um den Schüler:innen die Gradzahlen einer Drehung zu erarbeiten. Der Computer liebt Zahlen. Deshalb müssen wir uns eine Übersetzung von der Weite der Drehung in die Zahlenwelt überlegen. Eine komplette Umdrehung entspricht  $360^\circ$ . Jetzt sind die grauen Zellen gefragt: Wenn eine ganze Drehung  $360^\circ$  ist, wie viel sind dann eine halbe und eine viertel Drehung?

Wenn die Kinder das herausgefunden haben, zeichne die Erkenntnisse als Gedankenstütze an die Tafel. Nun kann das Spiel mit neuen Befehlen weitergespielt werden:


- Mache eine Drehung um 90° nach rechts.
- Mache eine Drehung um 90° nach links.
- Mache eine Drehung um 180°.
- Mache eine Drehung um 360°.

Als Hilfestellungen kannst Du die Kinder daran erinnern, dass sie die Begriffe **ganz/voll**, **halb** und **viertel** schon von der Uhrzeitansage oder vom Schneiden einer Pizza oder eines Kuchens kennen. Außerdem kannst Du zwei ausgedruckte Hände an die Tafel hängen, die den Kindern dabei helfen, links und rechts zu unterscheiden. Kopiervorlagen dazu findest Du im Begleitmaterial.



### Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen sie das Level **Esuri und die Birne** programmieren, bei dem die Raupe **Esuri** zu der Birne geführt werden muss, damit sie zum Schmetterling werden kann.

Bau nun mit den Schüler:innen gemeinsam die Tablets, Laptops oder Computer auf und stell sicher, dass jedes Gerät mit dem Internet verbunden ist. Für eine erleichterte Bedienung im Unterricht, öffnen die Schüler:innen das Level im Cubi-Editor über die Seite [editor.i4k.org](http://editor.i4k.org). Danach klicken sie erst auf **Menü** und dann auf **Öffnen** . Dort finden sie eine Liste mit allen Leveln. Über **Module** oben im Reiter gelangen sie zu den Leveln der Modulreihe. Die Schüler:innen müssen zum richtigen Level scrollen, um dieses zu öffnen. Gegebenenfalls bietet es sich an, gemeinsam mit den Kindern zu wiederholen, wie ein Programm aus den verschiedenen Bausteinen zusammengepuzzelt wird und wie sie das Level ausführen und zurücksetzen können. Arbeiten die Schüler:innen an Tablets, können sie den QR-Code einscannen, der sie direkt zu der Vorlage des Levels führt. Eine Kopiervorlage für den QR-Code findest Du im Begleitmaterial.

Ruf den Schüler:innen den Arbeitsauftrag in Erinnerung, indem ein Kind den Auftrag wiederholt.



Levelvorlage:  
[level.i4k.org/esuri\\_birne](http://level.i4k.org/esuri_birne)



Levellösung:  
[level.i4k.org/esuri\\_birne\\_lsg](http://level.i4k.org/esuri_birne_lsg)

## Arbeitsphase

Die Schüler:innen bearbeiten das Level **Esuri und die Birne**. Hier helfen sie Raupe Esuri durch das Ringelblumenbeet zur Birne.

Als erstes wird der Startbaustein **Wenn Start geklickt wurde** benötigt. Nun brauchen sie den Baustein **Drehe links um 90 Grad**, damit Esuri in die richtige Richtung guckt um loszulaufen. Dafür wird der Baustein **Gehe 100 Schritte** benötigt. So gelangt Esuri zur zweiten Ecke, wo der Baustein **Dre-**

**he rechts um 90 Grad** gebraucht wird. Jetzt befindet sich Esuri auf der Zielgeraden. Mit **Gehe 400 Schritte** gelangt sie zur Birne und kann endlich zum Schmetterling werden.



Als Differenzierungsaufgabe können die Kinder versuchen, dass Esuri sich nur links- oder rechtsherum dreht. Dafür müssen sie bei einem **Drehe**-Baustein den Wert **270** eingeben. Eine andere Idee wäre, den Schüler:innen aufzutragen, den schnellsten Weg zu finden – schaffen sie es, Esuri diagonal laufen zu lassen?



## Präsentation und Reflexion

Für die Präsentation treffen sich alle wieder im Tafelkino. Dort können die Schüler:innen ihre Arbeitsergebnisse vorstellen und darauf eingehen, was gut geklappt hat oder was ihnen schwergefallen ist. Wird mit Tablets gearbeitet, können die Schüler:innen diese mit nach vorne nehmen und den anderen Kindern zeigen, wie ihre Lösungen aussehen. Ist eine digitale Tafel oder eine ähnliche Präsentationsmöglichkeit vorhanden, können einzelne Lösungswege dort in groß gezeigt werden.

## Ausblick

Um die Inhalte dieses Moduls weiter aufzugreifen, bietet sich die Lektion **Code-Detektiv: Iruse und die Erdbeere** an. Dort müssen die Schüler:innen auf Fehlersuche gehen und in einem vorgegebenen Programm herausfinden, warum Raupe Iruse die Erdbeere nicht erreicht.

In der **Zusatzlektion Synchronsprechen** wird das Programmieren einer Sequenz als Werkzeug eingesetzt, um einen kurzen Dialog zu programmieren, welchen die Kinder am Ende der Klasse vortragen. Dadurch üben sie sich nicht nur im Programmieren sondern auch im Sprachgebrauch.

Das Modul **Sequenzen** kann an dieser Stelle abgeschlossen werden. Die Kinder können das Gelernte mit dem Arbeitsblatt **Handout Sequenzen** sichern sowie über ihren Lernstand mithilfe des Arbeitsblatts **Selbst-Check Sequenzen** reflektieren und so ihre Cubi-Mappe erweitern. Mehr dazu findest Du im Kapitel **Modulabschluss**.



# Code-Detektiv: Iruse und die Erdbeere

## Einführung

Bei der Lektion **Code-Detektiv** steht die Fehlersuche im Vordergrund. Das Suchen, Identifizieren und Beheben von Fehlern ist ein fundamentaler Bestandteil des Programmierens. Deshalb ist es essentiell, von Anfang an eine positive Fehlerkultur zu entwickeln. In der Informatik nennt man einen Fehler auch **Bug**, da einer der ersten scheinbar unerklärlichen Computerfehler 1945 entstand, weil eine Motte in den Rechner gekrochen war und dort einen Schalter blockierte.

Fehler oder Bugs sind keineswegs Indizien für Versagen. Vielmehr handelt es sich um eine Anomalie in einem Programm, eine Unregelmäßigkeit oder Regelwidrigkeit. Dem liegt die Sichtweise zugrunde, dass ein Programm nicht in sich falsch ist, sondern meist nur ein anderes Verhalten zeigt, als erwartet. Das Programm löst eine andere Aufgabe als ursprünglich geplant.

Diese Sichtweise wird beim **Code-Detektiv** aufgegriffen, indem die Schüler:innen einen vorprogrammierten Code erhalten, bei dem die Ursache dafür gefunden werden muss, warum er nicht bis zum Ende durchgeführt werden kann.

## Lernziele

- Die Schüler:innen vertiefen ihr Wissen über Sequenzen.
- Die Schüler:innen setzen sich aktiv und bewusst mit der Fehlersuche in einem Programm auseinander.
- Die Schüler:innen leben eine positiven Fehlerkultur.

## Geschichte des Levels

Raupe Iruse ist ganz schön neidisch auf Erusi. Iruse will auch so tolle Flügel haben und durch die Gegend fliegen können! Da hilft nur eins: Die Erbeere, die ganz in der Nähe liegt. Doch irgendwie geht auf dem Weg immer etwas schief... Wo wohl der Fehler liegt?

## Tabellarischer Unterrichtsverlaufsplan

| Zeit   | Phase                        | Unterrichtsschritte   | SF    | Material  |
|--|------------------------------|---|-------|---|
| 5  | Rückblick und Einstieg       | Rückblick auf die bisherigen Lektionen mit Fokus auf Schwierigkeiten und Probleme | T     |   |
| 15   | Erarbeitung                  | Comic <b>Fehlersuche</b>  | T     | Comic <b>Fehlersuche</b> , Präsentationstechnik |
| 5  | Übergang in die Arbeitsphase | Arbeitsauftrag erläutern, Tablets austeilern und Level öffnen                     | P     | Tablets/Laptops/PCs, AB <b>Meine Ente Erna</b>  |
| 15   | Arbeitsphase                 | Bearbeitung des Levels <b>Isure und die Erdbeere</b>                              | EA/PA | Tablets/Laptops/PCs, AB <b>Meine Ente Erna</b>  |
| 5  | Präsentation und Reflexion   | Vorstellen der Arbeitsergebnisse, Thematisierung von Entdeckungen und Problemen   | T     | Tablets/Laptops/PCs, Präsentationstechnik       |
| SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit |                              |   |       |   |

## Inhalte des Unterrichtsverlaufsplan

### Rückblick und Einstieg

Es wird sich im Tafelkino getroffen. Dort erinnern sich die Schüler:innen an die vorherigen Lektionen und aktivieren ihr Vorwissen. Thematisiere, dass es in dieser Stunde um eine Fehlersuche gehen wird und lass die Kinder über Schwierigkeiten und Probleme nachdenken, die sie bis hierhin beim Programmieren hatten. Frage die Kinder, was ihnen bei diesen Hürden geholfen hat.

### Erarbeitung


Lies gemeinsam mit den Schüler:innen das Comic **Fehlersuche**. Du findest es im Begleitmaterial. Mache nach der ersten Seite (sechstes Fenster) eine Pause und lass die Kinder Entes Tipp nachgehen, das Problem von Cubi und dem Detektivhund einmal laut zu erklären. Wenn sie sich des Problems bewusst geworden sind und sich erste Lösungsansätze überlegt haben, lies den Rest des Comics mit ihnen.

Überlege anschließend gemeinsam mit den Schüler:innen, warum das laute Nachdenken über das Problem geholfen hat, es zu lösen. Das Erklären des Problems hilft, die eigenen

Gedanken zu sortieren, wobei oft auffällt, wo sich der Fehler eingeschlichen hat. Dieser Ansatz der Fehlersuche ist in der Informatik fest etabliert und nennt sich **Rubberducking**.

## Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen sie sich das Level **Isure und die Erdbeere** angucken, herausfinden, wo der Fehler liegt und diesen korrigieren. Teile das Arbeitsblatt **Meine Ente Erna** aus. Sie hilft den Schüler:innen bei der Fehlersuche, indem durch sie anfängliche Scheu, das eigene Programm laut vorzulesen, abgelegt wird.

Bau nun mit den Schüler:innen gemeinsam die Tablets, Laptops oder Computer auf und stell sicher, dass jedes Gerät mit dem Internet verbunden ist. Für eine erleichterte Bedienung im Unterricht, öffnen die Schüler:innen das Level im Cubi-Editor über die Seite [editor.i4k.org](http://editor.i4k.org). Danach klicken sie erst auf **Menü** und dann auf **Öffnen** . Dort finden sie eine Liste mit allen Leveln. Über **Module** oben im Reiter gelangen sie zu den Leveln der Modulreihe. Die Schüler:innen müssen zum richtigen Level scrollen, um dieses zu öffnen. Arbeiten die Schüler:innen an Tablets, können sie den QR-Code einscannen, der sie direkt zu der Vorlage des Levels führt. Eine Kopiervorlage für den QR-Code findest Du im Begleitmaterial.

Ruf den Schüler:innen den Arbeitsauftrag in Erinnerung, indem ein Kind den Auftrag wiederholt.



Levelvorlage: [level.i4k.org/cd\\_iruse](http://level.i4k.org/cd_iruse)



Levellösung:  
[level.i4k.org/cd\\_iruse\\_lsg](http://level.i4k.org/cd_iruse_lsg)

## Arbeitsphase

Die Schüler:innen bearbeiten das Level **lruse und die Erdbeere**. Hier müssen sie herausfinden, warum Raupe Iruse es nicht zur Erdbeere schafft. Dabei wenden sie die Technik **Rubberducking** an, indem sie Ente Erna das Problem erläutern.

Der Fehler im Programm liegt dabei bereits im ersten **Bewegungs**-Baustein. Isure geht zu viele Schritte, sodass sie beim zweiten **Drehe rechts um 90 Grad**-Baustein mit dem Hinterteil an die Ringelblumen stößt. Werden die Schritte auf **250** redu-



ziert, wird das Problem behoben. Nun muss sie mithilfe des Bausteins **Gehe 300 Schritte** noch bis zur Erdbeere krabbeln und kann dann endlich zum Schmetterling werden.



Als Zusatzaufgabe können schnelle Schüler:innen ihre Ente Erna ausmalen. Das Comic kann als Vorlage dienen, Du kannst den Kindern aber auch gerne freistellen, wie ihre Ente Erna aussehen soll. Vielleicht hat Erna manchmal auch ein ganz buntes Federkleid!

## Präsentation und Reflexion

Für die Präsentation treffen sich alle wieder im Tafelkino. Dort können die Schüler:innen ihre Arbeitsergebnisse vorstellen und darauf eingehen, was gut geklappt hat oder was ihnen schwergefallen ist. Wird mit Tablets gearbeitet, können die Schüler:innen diese mit nach vorne nehmen und den anderen Kindern zeigen, wie ihre Lösungen aussehen. Ist eine digitale Tafel oder eine ähnliche Präsentationsmöglichkeit vorhanden, können einzelne Lösungswege dort in groß gezeigt werden.

## Ausblick

Die Lektion **Code-Detektiv: Iruse und die Erdbeere** ist die letzte des Moduls **Sequenzen**. In der Zusatzlektion **Synchronsprechen** wird das Programmieren einer Sequenz als Werkzeug eingesetzt, um einen kurzen Dialog zu programmieren, welchen sie am Ende der Klasse vortragen. Dadurch üben sich die Kinder nicht nur im Programmieren sondern auch im Sprachgebrauch.

Das Modul **Sequenzen** kann an dieser Stelle abgeschlossen werden. Die Kinder können das Gelernte mit dem Arbeitsblatt **Handout Sequenzen** sichern sowie über ihren Lernstand mithilfe des Arbeitsblatts **Selbst-Check Sequenzen** reflektieren und so ihre Cubi-Mappe erweitern. Mehr dazu findest Du im Kapitel **Modulabschluss**.



# Modulabschluss: Sequenzen

## Einführung

Der Modulabschluss dient der Sicherung des Gelernten. Hier haben die Schüler:innen Gelegenheit, ihre Erkenntnisse schriftlich festzuhalten und über sie zu reflektieren. Der Modulabschluss zeichnet sich primär durch die Erweiterung der Cubi-Mappe aus, weshalb sie in der Regel keine vollständige Unterrichtsstunde einnimmt. Die Arbeitsmaterialien sind so konzipiert, dass Du den Modulabschluss bereits nach dem Basislevel durchführen kannst, da sich nur Knobelaufgaben auf das Vertiefungslevel beziehen.

## Lernziele

- Die Schüler:innen haben die Möglichkeit, ihr Wissen über Sequenzen zu festigen und zu dokumentieren.
- Die Schüler:innen setzen sich aktiv mit ihrem Lernprozess auseinander und reflektieren über diesen.
- Die Schüler:innen erweitern ihre Cubi-Mappe und werden sich dessen Nutzen für zukünftige Projekte bewusst.



## Tabellarischer Unterrichtsverlaufsplan

| Zeit | Phase                        | Unterrichtsschritte                                | SF | Material   |
|------|------------------------------|--|----|--|
| 5    | Rückblick                    | Rückblick auf das Modul                            | P  |  |
| 5    | Übergang in die Arbeitsphase | Arbeitsauftrag erläutern, Arbeitsblätter austeilen | P  | AB <b>Handout Sequenzen</b> , AB <b>Selbst-Check Sequenzen</b> |
| 15   | Arbeitsphase                 | Bearbeitung der Arbeitsblätter                     | EA | AB <b>Handout Sequenzen</b> , AB <b>Selbst-Check Sequenzen</b> |
| 5    | Abschluss und Ausblick       | Das Modul wird abgeschlossen                       | P  |  |

SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit

## Inhalte des Unterrichtsverlaufsplans

### Rückblick

Leite ein Unterrichtsgespräch an, in dem sich die Schüler:innen kurz an die Inhalte des Moduls erinnern. Mit folgenden Kernaspekten haben sich die Schüler:innen im Modul beschäftigt:

- Analoger Einstieg:** Programmierung im Alltag.  
Sequenzielle Anweisungen geben und befolgen.
- Basislevel:** Die Lernsoftware **Cubi** kennenlernen und bedienen.  
Die Lernsoftware **Cubi** nutzen, um zu programmieren.
- Vertiefunslevel:** Bedeutung einer Drehung um 90°.
- Code-Detektiv:** Fehler in einem Code suchen und beheben.

### Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen sie die Arbeitsblätter **Handout Sequenzen** und **Selbst-Check Sequenzen** bearbeiten.

Mithilfe des Arbeitsblatts **Handout Sequenzen** können die Kinder ihre Erkenntnisse aus dem Modul festhalten. So können sie es in späteren Modulen oder während anderer Unterrichtseinheiten, in denen programmiert wird, als Gedankenstütze und Hilfsmittel heranziehen.

Mit dem Arbeitsblatt **Selbst-Check Sequenzen** können sich die Schüler:innen ihren Wissenstand bewusst machen. Es regt zum Reflektieren über den Lernprozess an und hilft

dabei, eigene Stärken und Herausforderungen zu identifizieren.

Solltest Du den Modulabschluss bereits nach dem Basislevel durchführen, mache die Schüler:innen darauf aufmerksam, dass sich die Knobelaufgabe des Handouts und die Aussage **Ich weiß, was eine Drehung um 90 Grad ist.** des Selbst-Checks auf ein Level beziehen, das die Kinder noch nicht kennen. Damit die Schüler:innen von der Aussage auf dem Selbst-Check nicht aufgehalten werden, kannst Du sie mit ihnen gemeinsam durchstreichen.

Die beiden Arbeitsblätter geben auch Dir eine Einsicht darüber, wie die Schüler:innen bei dem Modul mitgekommen sind. Kopiervorlagen und Lösungen sind im Begleitmaterial zu finden.

Ruf den Schüler:innen den Arbeitsauftrag in Erinnerung, indem ein Kind den Auftrag wiederholt.

### Arbeitsphase

Die Schüler:innen bearbeiten die Arbeitsblätter **Handout Sequenzen** und **Selbst-Check Sequenzen** am Platz in Einzelarbeit. Achte darauf, dass die Kinder die Blätter alleine bearbeiten, denn so können sie besser über die gelernten Inhalte des Moduls nachdenken und reflektieren. Schüler:innen die sich mit den Inhalten unsicher fühlen, hilft es, beim Selbst-Check für sich zu sein. So können sie die Aussagen ehrlich beantworten, ohne sich gegebenenfalls vor ihren Mitschüler:innen zu schämen.



### Abschluss und Ausblick

Schließ das Modul **Sequenzen** gemeinsam mit den Schüler:innen ab. Lass die Kinder mit der Klasse teilen, was ihnen gut gefallen hat und was sie nicht so sehr mochten. Wenn Du ein weiteres Modul mit der Klasse durchführen möchtest, gib den Schüler:innen einen Ausblick über dessen Inhalte. **Modul 2: Schleifen** eignet sich besonders, um an die bisherigen Inhalte anzuknüpfen.

Ihr habt in diesem Modul viel gelernt – Applaus! 🎉



## Zusatzlektion: Synchronsprechen

**Hinweis:** Diese Lektion dauert circa 90 Minuten. Sie ist für die Förderung der Sprachbildung konzipiert. Sie greift die Inhalte der vorherigen Lektionen auf und vertieft diese. Insbesondere muss sie nicht im selben Fach wie das Basis- und Erweiterungslevel umgesetzt werden.

### Einführung

Das Programmieren ist ein Werkzeug, welches in vielen Professionen, aber besonders in technischen Berufen, häufig genutzt wird. In dieser Zusatzlektion des Moduls **Sequenzen** ist das Programmieren nicht länger Lerngegenstand sondern Gestaltungswerkzeug. Die Schüler:innen nutzen ihre Programmierfähigkeiten, um ein eigenes Medienprodukt zu erstellen.

Dieses Medienprodukt ist ein digitales Theaterstück. Mithilfe der bekannten **Start-** und **Bewegungs-**Bausteine und neuen Befehls-Bausteinen aus den Kategorien **Aussehen** (Sprechblasen) und **Kontrolle** (warten) realisieren die Schüler:innen in Partnerarbeit oder kleineren Gruppen ein selbstgeschriebenes Drehbuch, welches sie zum Abschluss ihren Mitschüler:innen vorstellen.

Dabei werden zum einen die Planungskompetenzen der Schüler:innen, zum anderen ihre Kreativität und schauspielerischen Sprachfähigkeiten herausgefordert. Die Schüler:innen programmieren einen kurzen Dialog von zwei Figuren. Bei der Vorstellung leihen die Schüler:innen diesen ihre Stimmen und vertonen so das Theaterstück.

### Lernziele

- Die Schüler:innen wiederholen die Programmierung mit bekannten **Start-** und **Bewegungs-**Bausteine.
- Die Schüler:innen können ein Medienprodukt selbstständig planen, umsetzen und überarbeiten.
- Die Schüler:innen können ihre Ideen selbstbewusst in der Gruppenarbeitsphase einbringen und begründen.
- Die Schüler:innen können Ideen ihrer Mitschüler:innen in der Gruppenarbeitsphase tolerierend zur Kenntnis nehmen und in Bezug auf deren Realisierbarkeit abwägen.

- Die Schüler:innen können ihr Medienprodukt der Klasse präsentieren und Designentscheidungen begründen.
- Die Schüler:innen können fundiertes und konstruktives Feedback zu den kreativen Leistungen ihrer Mitschüler:innen geben.

## Tabellarischer Unterrichtsverlaufsplan

| Zeit | Phase                        | Unterrichtsschritte   | SF | Material                                     |
|------|------------------------------|---|----|--|
| 5    | Einstieg                     | Sammeln von Dialogsituationen   | P  |  |
| 10   | Erarbeitung                  | Kennenlernen der neuen Bausteine  | T  | Präsentationstechnik                         |
| 10   | Übergang in die Arbeitsphase | Ablauf des Projekts <i>Synchronsprechen</i>   | P  | Präsentationstechnik                         |
| 45   | Arbeitsphase                 | Planung des Dialogs, Umsetzung in Lernsoftware Cubi, Anpassung des Drehbuchs, Üben der Präsentation | PA | AB <b>Unser Dialog</b> , Tablets/Laptops/PCs |
| 20   | Präsentation und Reflexion   | Vorstellung der digitalen Theaterstücke   | T  | Tablets/Laptops/PCs, Präsentationstechnik    |

SF = Sozialform, P = Plenum, S = Sitzkreis, T = Tafelkino, EA = Einzelarbeit, PA = Partnerarbeit, GA = Gruppenarbeit

## Inhalte des Unterrichtsverlaufsplans

### Einstieg

Steig in die Stunde ein, indem Du mit der Klasse Beispiele für vertraute Dialogsituationen sammelst, z.B. sich vorstellen, wenn man jemand Neues trifft, von einem Hobby erzählen, sich begrüßen/verabschieden. Zwei Schüler:innen, die sich freiwillig melden, suchen sich eine Situation aus und improvisieren einen kurzen Dialog mit vier bis fünf Wortwechsel zu dem Thema.

### Erarbeitung

Es wird sich im Tafelkino getroffen. Öffne das Level Synchronsprechen im Cubi-Editor über [editor.i4k.org](http://editor.i4k.org) oder den QR-Code, sodass alle Schüler:innen es sehen können.

Die Schüler:innen erinnern sich an vergangene Programmierprojekte und aktivieren ihr Vorwissen. Lass ein Kind beschreiben, wie Du den **Start**-Baustein auf die Programmierfläche ziehst und das Level startest und zurücksetzt.

Frage die Klasse, welche Kategorien neu für sie sind. Probiert die neuen **Sage**- und **Warte**-Bausteine aus, indem ihr ein kurzes Programm wie hier implementiert.

Programmiert gemeinsam den Anfang des improvisierten Dialogs aus dem Einstieg. Weise darauf hin, dass Figur B warten muss, wenn Figur A etwas sagt, um ihr nicht ins Wort zu fallen. Hier kann es helfen, nicht erst das Programm von Figur A fertig zu schreiben und dann das Programm von Figur B zu schreiben, sondern immer nur einen **Sage**-Baustein zu ergänzen und dann gleich bei der anderen Figur den entsprechenden **Warte**-Baustein hinzuzufügen.

Stelle sicher, dass alle Schüler:innen den Zweck der neuen Bausteine erkannt haben und das Prinzip von sich abwechselnden **Sage**- und **Warte**-Bausteinen nachvollziehen können.

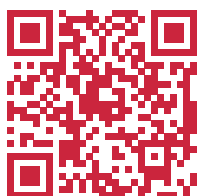


## Übergang in die Arbeitsphase

Besprich mit den Schüler:innen den Arbeitsauftrag der nachkommenden Arbeitsphase. Hier sollen die Schüler:innen zunächst einen eigenen Dialog auf Papier aufschreiben. Er sollte für genau zwei Figuren sein und aus mindestens drei Wortmeldungen pro Figur bestehen. **Hinweis:** Sollte es Dreiergruppen geben, kann ein Kind bei der Vorstellung die Erzählrolle übernehmen und den Dialog einführen und abschließen. Zeige das Arbeitsblatt **Unser Dialog** und besprich mit der Klasse, wie die Tabelle bei Aufgabe 3 auf dem zweiten Arbeitsblatt ausgefüllt werden soll.

Im nächsten Schritt der Arbeitsphase sollen die Schüler:innen diesen Dialog in Cubi nach dem Muster der Erarbeitungsphase umsetzen. Schließlich sollen sie ihr Level testen, den Figuren ihre Stimme geben und die Vorstellung proben.

Ruf den Schüler:innen den ersten Schritt der Arbeitsphase in Erinnerung, indem ein Kind den Auftrag wiederholt. Teile anschließend die Gruppen für die Arbeitsphase ein und verteile die Arbeitsblätter **Unser Dialog**.



Levelvorlage:

[level.i4k.org/synchronsprechen](http://level.i4k.org/synchronsprechen)




Levellösung:

[level.i4k.org/synchronsprechen\\_lsg](http://level.i4k.org/synchronsprechen_lsg)

## Arbeitsphase

Gehe durch den Klassenraum und unterstütze die Schüler:innen bei Fragen. Achte darauf, dass der Dialog nicht zu lang wird und sie nach maximal der Hälfte der Zeit mit dem Programmieren starten.

Bau für die Programmierung mit den Schüler:innen gemeinsam die Tablets, Laptops oder Computer auf und stell sicher, dass jedes Gerät mit dem Internet verbunden ist. Pro Gruppe benötigen die Schüler:innen ein Gerät. Für eine erleichterte Bedienung im Unterricht, öffnen die Schüler:innen das Level im Cubi-Editor über die Seite [editor.i4k.org](http://editor.i4k.org). Danach klicken sie erst auf **Menü** und dann auf **Öffnen** . Dort finden sie eine Liste mit allen Leveln. Über **Module** oben im Reiter gelangen sie zu den Leveln der Modulreihe. Die Schüler:innen müssen zum richtigen Level scrollen, um dieses zu öffnen. Arbeiten die Schüler:innen an Tablets, können sie den QR-Code einscannen, der sie direkt zu der Vorlage des Levels führt. Eine Kopiervorlage für den QR-Code findest Du im Begleitmaterial.

Wenn noch Zeit ist, können die Schüler:innen den Figuren ein anderes Kostüm anziehen. Dafür klicken sie auf das große Plus unterhalb der Bühne, öffnen die Kostümbibliothek und wählen ein Kostüm ihrer Wahl aus. Um das Kostüm der Figur zu wechseln, wählen sie das entsprechende Kostüm in der Kostümauswahl der Figur an.

## Präsentation und Reflexion

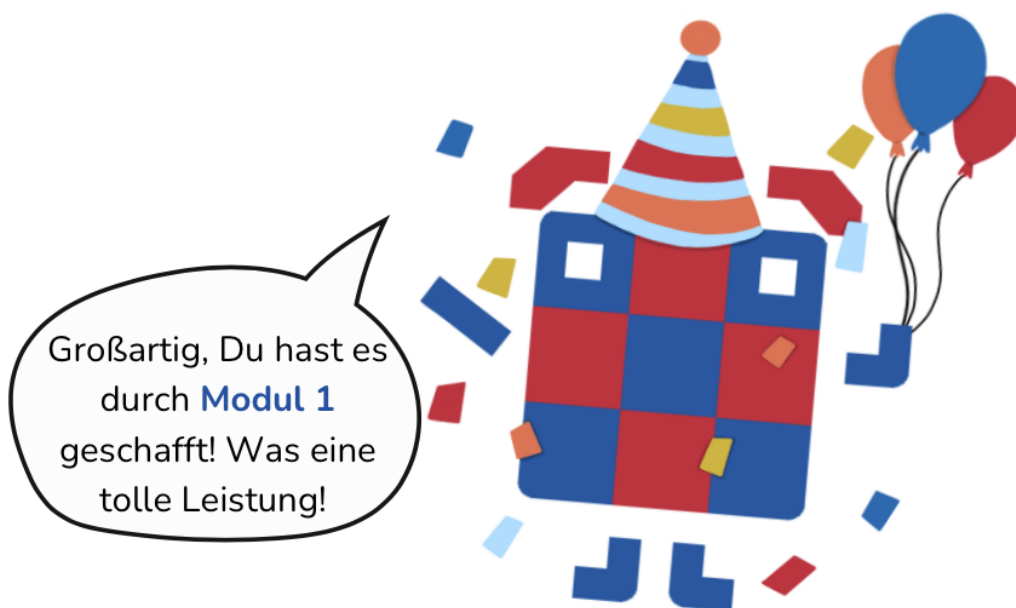
Für die Präsentation treffen sich alle im Tafelkino. Dort stellen die Schüler:innen ihre Arbeitsergebnisse vor, indem sie den Dialog nachsprechen während das Level läuft. Geht darauf ein, was gut geklappt hat oder was schmergefallen ist. Ihre Mitschüler:innen können im Anschluss fundiertes und konstruktives Feedback zu den Produkten geben. Achte darauf, dass die Kinder genau beschreiben, was ihnen (nicht) gefallen hat und hake im Zweifel nochmal nach.

Ist eine digitale Tafel oder ähnliche Präsentationstechnik vorhanden, können die digitalen Theaterstücke dort in groß gezeigt werden. Einen zusätzlichen Effekt hat es, wenn die Bühne in Cubi größer gezogen wird. Dazu kannst Du den Mittelbalken zwischen Programmierfläche und Bühne drücken und beim Gedrückthalten nach links oder rechts ziehen.

## Abschließende Bemerkungen

Der Inhalt dieser Lektion kann in verschiedenen Fächern mit unterschiedlichen Schwerpunkten aufgegriffen und wiederholt werden. Die Lektion zeichnet sich dadurch aus, dass die Schüler:innen ihr Wissen über ein Thema unter Einsatz von digitalen Medien kreativ und mündlich darstellen können. Sie erfahren, dass sie aktive Gestalter:innen und nicht passive Konsument:innen von Medien sind.

## Geschafft!





# Baustein-Lexikon

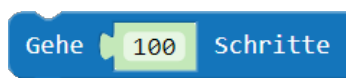
## Start

Der erste Baustein eines Blocks aus mehreren Bausteinen ist immer ein **Start**-Baustein. Ein Programm einer Figur kann beliebig viele **Start**-Bausteine haben. **Start**-Bausteine zeichnen sich durch die Rundung am oberen Teil aus. Diese sagt aus, dass **Start**-Bausteine nicht an andere Bausteine angehängt werden können.



Der **Start**-Baustein **Wenn Start geklickt wurde** ist der erste Baustein, den die Schüler:innen kennenlernen. Nachfolgende Bausteine werden nacheinander ausgeführt, unmittelbar nachdem das Level gestartet wurde.

## Bewegung



Der Baustein **Gehe ... Schritte** bewegt die Figur die entsprechende Anzahl an Pixel in die aktuelle Richtung der Figur. Im Normalfall ist dies bei Programmstart nach rechts.



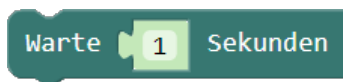
Mit dem Baustein **Drehe rechts/links um ... Grad** dreht sich die Figur in die ausgewählte Richtung um die entsprechende Gradzahl. In den ersten Level brauchen die Schüler:innen nur den rechten Winkel.

## Aussehen



Wenn der **Sage**-Baustein ausgeführt wird, erscheint für die Figur eine Sprechblase. Dort wird der Text angezeigt, welcher in das Textfeld geschrieben wurde. Nach der angegebenen Zeit verschwindet die Sprechblase und der nächste Baustein wird ausgeführt.

## Kontrolle



Gelangt ein Programm zu einem **Warte**-Baustein, dann bleibt es hier für die Anzahl der Sekunden stehen. Andere Programmteile der Figur, die ihren eigenen **Start**-Baustein haben, werden hierdurch nicht unterbrochen. Erst nach Ablauf der Zeit wird der nächste Baustein ausgeführt.