



# INFORMATIK AN GRUNDSCHULEN

**Handreichungen und Unterrichtsmaterial  
Hinweise zur Schulung/Fortbildung**

**Projekt  
Informatik an Grundschulen  
(IaG)**



Rheinisch-Westfälische Technische Hochschule Aachen  
Universität Paderborn  
Bergische Universität Wuppertal

Ministerium für  
Schule und Bildung  
des Landes Nordrhein-Westfalen





*Informatik an Grundschulen (IaG) – Einführung – Grundlagen*

*Ludger Humbert, Johannes Magenheim, Ulrik Schroeder, Martin Fricke, Nadine Bergner*

*16. März 2019*



**Inhaltsverzeichnis**

	Seite
1 Vorwort	GB 02
2 Informatik, Informatiksysteme und Grundschule?	GB 03
3 Was ist Informatik?	GB 04
4 Informatik – informatische Bildung	GB 05
5 Informatik als Wissenschaft – Fachgebiete – stabil seit 1976	GB 07
6 Informatische Bildung kommt in die Schule	GB 08
7 Zusammenfassung wesentlicher Argumente für Informatik an Grundschulen	GB 12

1 Vorwort

Liebe Grundschullehrerinnen und Grundschullehrer,  
sehr geehrte Damen und Herren,

es gehört zum Bildungsauftrag von Schule, Schülerinnen und Schüler bestmöglich auf das Lernen und Leben in einer sich rasant verändernden digitalisierten Welt vorzubereiten.

Hierzu zählen neben digitalen Anwendungskompetenzen und kritischer Medienkompetenz auch informatische Kompetenzen. Denn diese ermöglichen es, die grundlegende Funktionsweise von Informatiksystemen zu verstehen und kritisch zu hinterfragen.

Als Grundlage dafür hat Nordrhein-Westfalen im Jahr 2017 nach einem umfassenden Arbeitsprozess mit Wissenschaft und medienpädagogischer Fachperspektive den Medienkompetenzrahmen NRW entwickelt. Dieser trägt der im Dezember 2016 beschlossenen *Strategie der Kultusministerkonferenz »Bildung in der digitalen Welt«* Rechnung, die verbindliche Anforderungen an das Lehren und Lernen in der digitalen Welt beschreibt. Somit ist der Medienkompetenzrahmen NRW auf nationale und internationale Entwicklungen abgestimmt. Mit seinen sechs Kompetenzbereichen und insgesamt 24 Teilkompetenzen ermöglicht er eine systematische Medienbildung von der Primarstufe bis zum Ende der Pflichtschulzeit. Besonders hervorzuheben ist, dass mit dem Kompetenzbereich »Problemlösen und Modellieren« die Basis für eine informatische Grundbildung für alle Schülerinnen und Schüler gelegt wird – und diese soll bereits in der Grundschule beginnen.

Deshalb freue ich mich sehr, dass im Rahmen des Projekts »Informatik an Grundschulen« wichtige Impulse gegeben werden, wie Kompetenzen aus dem Bereich der Informatik im Primarbereich verankert werden können. Hierzu wurden die Module »Digitale Welt«, »Ich habe ein Geheimnis!« und »Wie funktioniert ein Roboter?« an Grundschulen unterrichtlich erprobt und auf Grundlage der Rückmeldungen der beteiligten Grundschul-

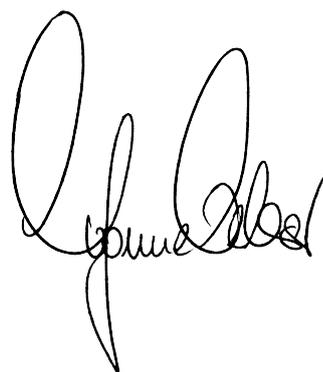
lehrkräfte weiterentwickelt. Alle Module sind »unplugged« konzipiert und erfordern deshalb keine besondere technische Ausstattung.

Die im Projektkontext erarbeiteten Handreichungen und Unterrichtsmaterialien zeigen Möglichkeiten auf, wie informatische Bildung in den Fächern der Primarstufe vermittelt werden kann. Dabei werden auch die fachlichen Hintergründe ausführlich erläutert und Anregungen zur konkreten unterrichtlichen Umsetzung gegeben. Zudem enthalten die Materialien Hinweise zum Einsatz in der Lehrerfortbildung.

Die Kooperation von Wissenschaft und Schule war eine wesentliche Gelingensbedingung für das Projekt. Für ihre engagierte Arbeit möchte ich den beteiligten Lehrkräften sowie den Expertinnen und Experten aus dem Bereich der Fachdidaktik Informatik von der Universität Paderborn, der Bergischen Universität Wuppertal und der RWTH Aachen herzlich danken. Mein Dank gilt ebenso dem Heinz Nixdorf MuseumsForum für die Begleitung des Projekts.

Ich wünsche Ihnen, dass Sie bei der Lektüre viele hilfreiche Anregungen und Ideen zur Weiterentwicklung Ihres Unterrichts gewinnen.

Ihre



**Yvonne Gebauer**  
Ministerin für Schule und Bildung  
des Landes Nordrhein-Westfalen

2 Informatik, Informatiksysteme und Grundschule?

**Einordnung des Themas – Informatik an Grundschulen**

- Welche Bedeutung hat das Thema für uns?
- Welche Bedeutung hat das Thema für die Kinder?
- Was ist der Alltagsbezug?

In vielen Dingen, die uns im Alltag umgeben, »steckt Informatik drin« ohne dass wir uns dessen bewusst sind. Bei Computern, Laptops und Smartphones mag es noch offensichtlich sein, dass es sich um Informatiksysteme handelt.

Informatiksysteme sind aber viel weiter verbreitet und kommen in fast allen unseren Lebensbereichen vor:

*im Haushalt* Fernseher, Haushaltsgeräte, Fotoapparate ...

*im Verkehr* Autos, Parkscheinautomaten, Nah- und Fernverkehrssysteme, Verkehrsampeln, Verkehrsleitsysteme, Aufzüge ...

*im Öffentlichkeits- und Freizeitbereich* Computerspiele, Spielautomaten, Digitale Medien, Soziale Netzwerke ...

*in der Medizin* Diagnose- und Therapiegeräte, Gesundheitskarte ...

*beim Einkaufen* Supermarktkassen, Zugangskontrollsysteme, Bankautomaten ...

Informatiksysteme haben offensichtlich äußerst vielfältige Erscheinungsformen, sind unterschiedlich komplex und oft miteinander vernetzt. Bei vielen dieser Geräte und Systeme hinterlassen wir Datenspuren, wenn wir sie benutzen. Eine zentrale Aufgabe der informatischen Bildung ist es daher, informatische Grundkenntnisse über die Funktionsweise dieser Systeme zu vermitteln, um so einen kompetenten und bewussten Umgang zu ermöglichen.

**Informatik ist überall dort, wo ...**

- Abläufe automatisiert gesteuert und geregelt werden
- Daten digital gespeichert und ausgegeben werden
- Daten übertragen werden
- Daten verändert und berechnet werden



Abbildung 1: Mikrowelle, Aufzug, Spielautomat, EEG, Diebstahlstele

Informatiksysteme – Eigenschaften und Zuschreibungen

- »allgegenwärtig«
- übertragen
- kontrollieren & regulieren
- verändern & berechnen
- speichern & ausgeben
- vernetzen

Informatiksysteme werden mittels Methoden der Informatik gestaltet. Informatische Methoden und Konzepte können nach Erscheinungsformen und Begegnungsmöglichkeiten durch drei Phänomenbereiche klassifiziert werden.

Phänomene → Phänomenbereiche

1. direkt mit Informatiksystemen
2. indirekt mit Informatiksystemen
3. ohne Informatiksysteme

Zuordnung der Phänomenbereiche an dem Beispiel Supermarkt:

1. Kassiererin tippt etwas in die Kasse ein
2. Ware wird über das Scannerfeld geführt
3. Kundin stellt sich an der Warteschlange an



Abbildung 2: TV, Ampel, Supermarktkasse

### 3 Was ist Informatik?

In einem Informatikschulbuch (Balzert, 1976) wird Informatik als zusammengesetzter Begriff definiert:

Informatik – der Begriff

Kompositum aus **I**nformation und **A**utomatik

Im deutschen Sprachraum wird die Bezeichnung Informatik erstmalig 1957 zur Charakterisierung der automatischen Verarbeitung von Daten verwendet. Wir gehen von der folgenden Definition aus:

Definition Informatik – Informatiksystem

Die Wissenschaft Informatik beschäftigt sich mit Strukturen und Prozessen, die in Form von Daten und Algorithmen repräsentiert werden (können). Diese Prozesse werden **informatisch modelliert**, so dass sie automatisch durchgeführt werden (können).

.....

Die Systeme, die dazu von der Informatik konstruiert werden, werden als **Informatiksysteme** bezeichnet. Einige Dinge, die Sie auf dem Schreibtisch, in der Tasche, in Ihrem Auto, ... haben, sind Informatiksysteme. Da Informatiksysteme meist von Menschen genutzt werden und sie in deren soziale Handlungskontexte eingebunden sind, werden sie auch als sozio-technische Informatiksysteme bezeichnet.

Es ist wichtig, deutlich zwischen den Begriffen **Daten** und **Information** zu unterscheiden:

**Daten ↔ Information**

Daten weisen immer eine Struktur auf: **Syntax**.  
 Bei Information kommt hinzu, dass Daten interpretiert werden (**Semantik**) und dies gegebenenfalls zu Aktionen führt (**Pragmatik**).

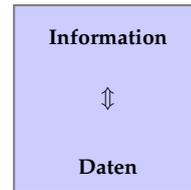


Abbildung 3: Daten und Information

**Beispiel Daten ↔ Information**

Daten müssen neben Zeichen auch eine Strukturinformation enthalten, damit sie von Informatiksystemen verarbeitet werden können:

- "24122017" kann zum Beispiel als Zeichenkette in einem Text definiert werden
- als die Zahl: 24 Millionen 122 Tausend 17
- als Datum: 24.12.2017
- das ist das Datum des nächsten Weihnachtsfestes oder der Hochzeitstag eines Paares oder der Tag, bis zu dem eine Person eine Aufgabe erledigen möchte.

.....

Während die Darstellung **a)–c)** formale Strukturelemente enthalten, und damit Anweisungen, wie sie von einem Informatiksystem verarbeitet werden müssen, enthält **d)** kontextuelle Hinweise, wie das Datum zu interpretieren ist. Diese semantische Interpretation ist in den Daten für das Informatiksystem so nicht enthalten, sondern basiert auf Angaben, über die sehr viele (Weihnachten) oder nur wenige Personen (Hochzeitstag) verfügen evtl. zunächst nur eine Person verfügt (Aufgabe).

## 4 Informatik – informatische Bildung

### Allgemeinbildung

Mit den Kriterien von Klafki (Klafki, 1993) für die allgemeine Bildung werden fachunabhängig grundlegende Kompetenzen beschrieben, die die allgemeine Bildung in demokratisch verfassten Gesellschaften ausweisen.

**Allgemeinbildung – nach Klafki**

<ul style="list-style-type: none"> <li>• Mitbestimmungsfähigkeit</li> <li>• Selbstbestimmungsfähigkeit</li> <li>• Solidaritätsfähigkeit</li> <li>• Bildung für alle</li> </ul>	<ul style="list-style-type: none"> <li>• Allseitige Bildung <i>bezogen auf grundlegende menschliche Fähigkeitsdimensionen</i></li> <li>• Bildung im Sinne des Allgemeinen <i>gesellschaftlicher Gesamtzusammenhang: epochaltypische Schlüsselprobleme, Gegenwarts- und Zukunftsbedeutung</i></li> </ul>
--	---

Die Aufgabe der allgemeinen Bildung besteht darin, dass Lernende auf ihrem Weg der Entwicklung zur mündigen Bürgerin resp. zum mündigen Bürger so begleitet werden, dass sie ihre eigene **Zukunft verantwortlich gestalten** können.

- Was ist allgemeinbildend an der Informatik?

Informatische Bildung leistet zur Entwicklung der Allgemeinbildung in der Informations- und Wissensgesellschaft einen wichtigen Beitrag. Sie soll Schülerinnen und Schüler befähigen, sich mit Problemen unserer durch Informatiksysteme und Digitalisierung geprägten Gesellschaft kompetent auseinanderzusetzen, und ihre erworbenen informatischen Kompetenzen erfolgreich sowohl für ihre künftige individuelle Lebensgestaltung als auch für eine durch Solidarität geprägte gesellschaftliche Entwicklung zu nutzen.

! **Digitalisierung** ist Teil der Informatik, **informatische Bildung** ist Teil der **Allgemeinbildung** → Bildung in der digitalen Welt

- Was sind – aus Sicht der Informatikdidaktik – die Alleinstellungsmerkmale der Informatik, die in keinem anderen Schulfach verfügbar gemacht werden können?

**Informatik**

- **Informatische Modellierung** bietet einen **Zugang zur Welt**, der so mit keiner anderen Wissenschaft möglich ist (vgl. Abbildung 4) – weder die Naturwissenschaften noch die Mathematik erlauben es, »Modellierungsszenarien« in lauffähige Informatiksysteme umzusetzen **und** zum Ablauf zu bringen (z. B. in Form von Programmen).
- In der Informatik werden **dynamische Prozesse** modelliert. Wir sind – auch in der Schule / im Kindergarten – in der Lage, diese Informatikmodelle »zum Laufen zu bringen«.
- Mit der **phänomenorientierten Informatikdidaktik** liegt ein Ansatz vor, durch den Alltagssituationen, in denen **keine Informatiksysteme** auftreten, mit der **Brille der Informatik** betrachtet werden. Diese Sicht ist als didaktischer Aufschluss gut für den Primarbereich geeignet.
- Ergebnisse der informatischen Modellierung verändern die Welt – um Gestaltungs- und Beurteilungskompetenz für zukünftige Anforderungen zu entwickeln, bedarf es der **theoriegeleiteten** Durchdringung der informatischen Modellierung auf einer fachlich ausgewiesenen Basis.
- Wesentliche Erkenntnisse der theoretischen Informatik bestehen darin, dass der informatischen Modellierung und der Gestaltung von Informatiksystemen immanent methodische und axiomatische Grenzen gesetzt sind, die nicht überwunden werden können. Diese Erkenntnisse sind in didaktisch verantwortlicher Form zu einem grundlegenden Bildungsgegenstand auszugestalten.

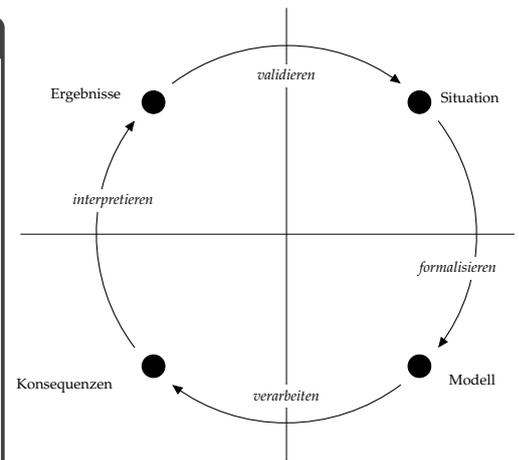


Abbildung 4: Der Informatische Modellierungskreis

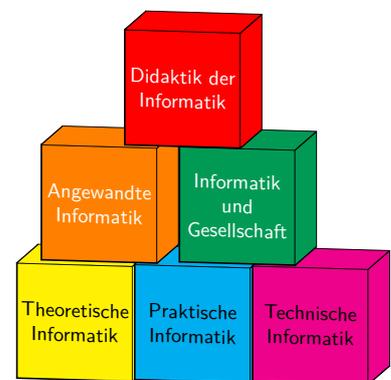


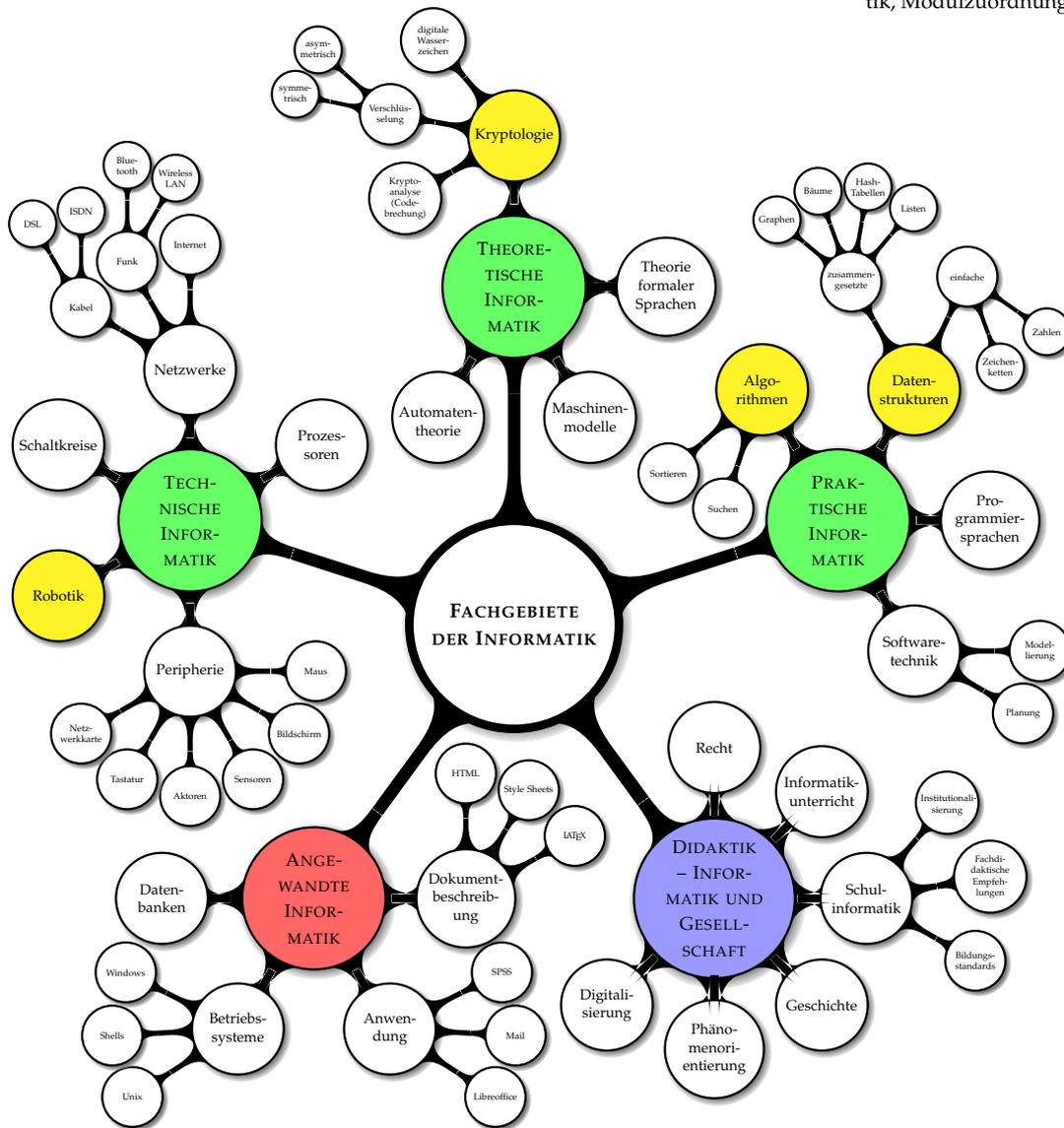
Abbildung 5: Fachgebiete der Informatik: Kerninformatik und mehr

5 Informatik als Wissenschaft – Fachgebiete – stabil seit 1976

Der Fakultätentag Informatik beschloss am 30. April 1976 die Strukturierung der Wissenschaft durch den Ausweis der folgenden sechs Fachgebiete (vgl. Abbildung 5):

- |                            |                         |                                |
|----------------------------|-------------------------|--------------------------------|
| 1. Technische Informatik   | } <b>Kerninformatik</b> | 4. Angewandte Informatik       |
| 2. Theoretische Informatik |                         | 5. Informatik und Gesellschaft |
| 3. Praktische Informatik   |                         | 6. Didaktik der Informatik     |

Abbildung 6: Fachgebiete der Informatik, Modulzuordnung



Die ersten drei Fachgebiete werden als **Kerninformatik** bezeichnet. In der Abbildung 6 (nach Humbert, 2018) sind die beiden Fachgebiete *Informatik und Gesellschaft* und *Didaktik der Informatik* in einem Knoten dargestellt. Den Fachgebieten werden auf der zweiten Ebene der Grafik wichtige Punkte zugeordnet, die zum Teil in einer dritten Ebene ausdifferenziert sind. Dabei sind Zuordnungen auf der zweiten Ebene nicht trennscharf, wie an den Beispielen Datenbanken und Betriebssysteme erkennbar ist, die teilweise enger an die praktische Informatik angebunden werden.

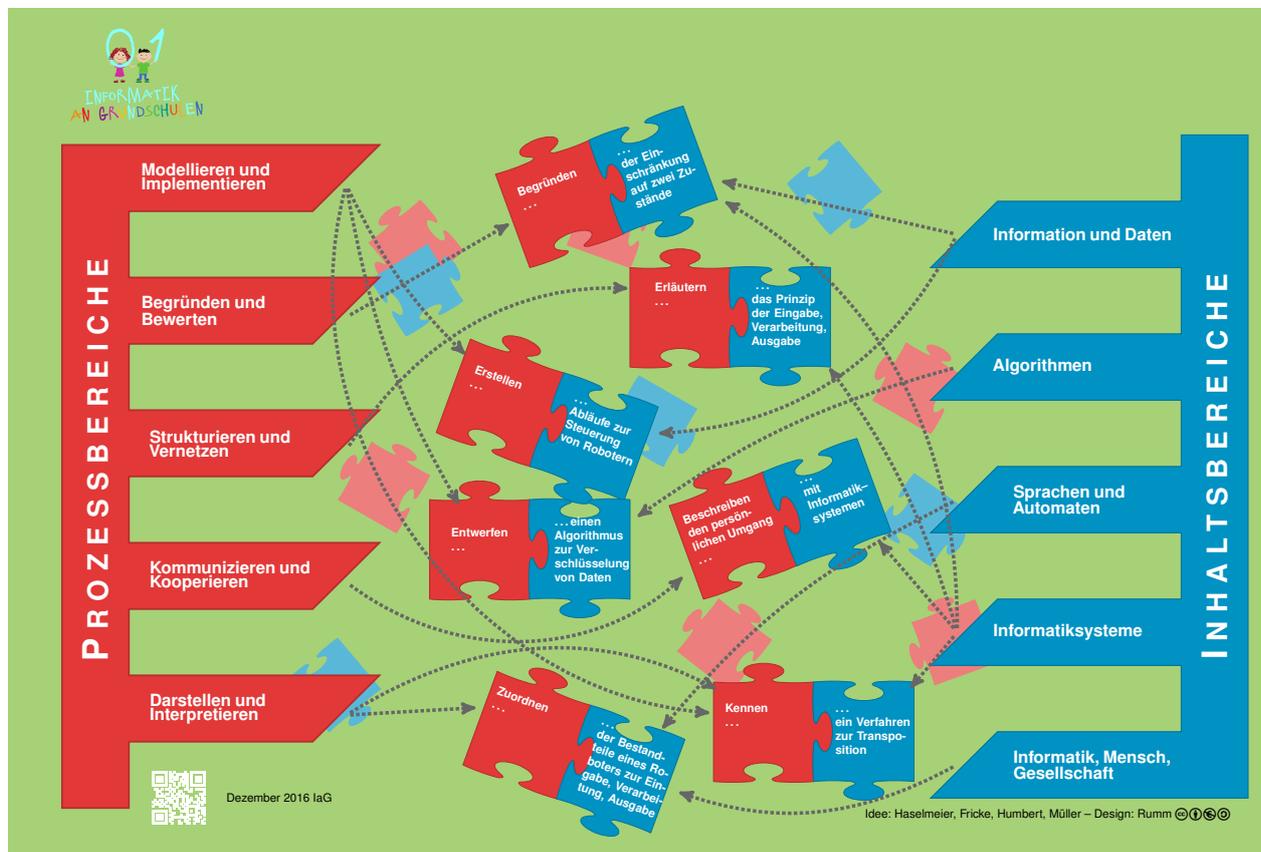
6 Informatische Bildung kommt in die Schule

Die Ergebnisse der informatischen Modellierung durchdringen und verändern **fast alle** Lebens- und Arbeitsbereiche, alle Wissenschaften, alle Schulfächer – ja die gesamte Arbeit auch **in** der Schule. Damit wird die informatische Modellierung als »Sicht auf die Welt« zu einem Schlüssel für das Weltverständnis. Für den kritisch-konstruktiven Umgang mit Ergebnissen der informatischen Modellierung bedarf es einer fachlich validen Basis.

6.1 Konkretisierung Informatische Bildung – kompetenzorientiert

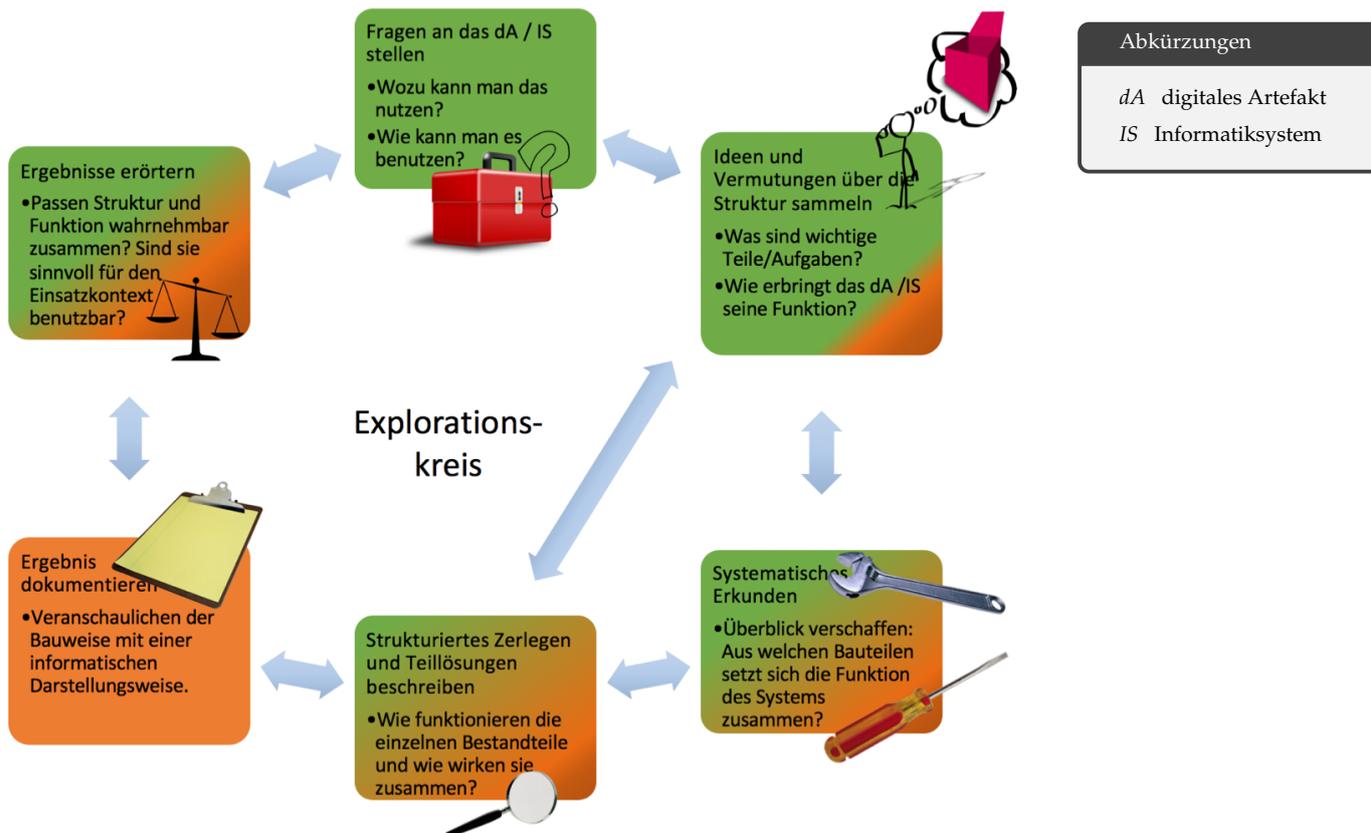
Um Klarheit darüber zu gewinnen, welche konkreten Kompetenzen die Kinder bereits im Grundschulalter entwickeln können – ja: entwickeln sollen – haben wir im Projekt »Informatik an Grundschulen« Prozesse und Inhalte der Informatik beispielhaft verwoben und Kompetenzen abgeleitet. Diese werden in Abbildung 7 auszugsweise dargestellt. Kompetenzen werden so in Bezug auf bestimmte Inhalte durch die Fähigkeit zu konkretem, problemlösendem Handeln sichtbar. Das Konzept folgt damit der Struktur der Informatiklehrpläne in NRW für die Sek I und die gymnasiale Oberstufe und bezieht sich auf die diesen zugrundeliegenden Empfehlungen der Gesellschaft für Informatik (GI). Die Kompetenzen orientieren sich an den Empfehlungen der GI zu den *Kompetenzen für informatische Bildung im Primarbereich* (Gesellschaft für Informatik e. V., 2019). In dem Beitrag »Der Medienkompetenzrahmen in Nordrhein-Westfalen« (vgl. Fricke und Schaumburg, 2018) wird gezeigt, dass die im Medienkompetenzrahmen geforderten informatikbezogenen Kompetenzen durch diese Empfehlungen untersetzt werden.

Abbildung 7: Informatik an Grundschulen – Kompetenzen



Im Projekt *Informatik an Grundschulen (IaG)* finden derartige fundamentale informatikbezogene Problemlöseprozesse ohne den Einsatz von Informatiksystemen statt. Da Kinder sich Wissen über ihre Welt meist spielerisch und explorierend aneignen, kann diese motivierende, enaktive Zugangsweise auch für den Umgang mit digitalen Artefakten (z. B. digitales Bild) und Informatiksystemen genutzt werden. Indem Schülerinnen und Schüler verschiedene Interaktionen mit einem Informatiksystem ausprobieren und dabei das Systemverhalten beobachten, können Interaktionsmuster erkannt und – basierend auf beobachtbaren Funktionen des Systems und lernförderlichen Impulsen der Lehrkräfte – erste rudimentäre Modelle über deren innere Struktur aufgebaut werden.

Abbildung 8: Grafik Explorationskreis

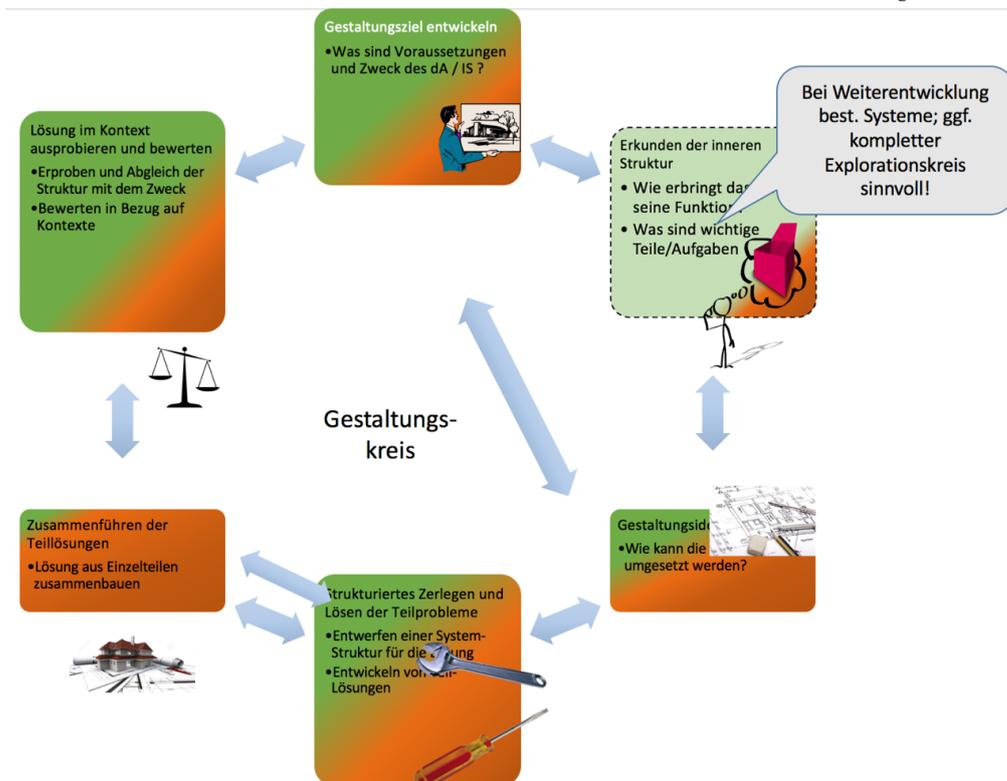


Die Exploration eines gegebenen Artefakts bzw. eines Informatiksystems kann als eine spezifische, besonders für Anfänger geeignete Variante des Modellierungskreises angesehen werden.

Bevor ein Informatiksystem neu gestaltet wird, kann ein bereits existierendes erkundet und so über seine äußerlich sichtbare Funktionen auf seine innere Bauweise geschlossen werden. Es findet ein Perspektivenwechsel statt: Von der beobachtbaren Funktion wird auf die mögliche innere Struktur des Informatiksystems geschlossen, von der Außensicht auf die Innensicht gewechselt, von der Exploration zur möglichen Gestaltung des Informatiksystems. Der Gestaltungskreis (Abbildung 9) ist eine spezifische Form des Modellierungskreises. Am Anfang steht zumeist ein Problem, das es mittels informatischem Modellierens möglichst zu lösen gilt. Noch stärker als beim Explorieren sind die Schritte ineinander verzahnt, sodass Zyklen und Rücksprünge als schrittweises Annähern an die gewünschte Lösung möglich sind. Da neben dem Gestalten eines komplett neuen Systems auch auf der Grundlage eines bestehenden Informatiksystems – das zunächst exploriert werden muss – aufgebaut werden kann, sind beide Kreise eng verbunden.

Beide Kreise wechseln zwischen zwei Perspektiven hin und her, verdeutlicht durch Farben: Rot bezieht sich auf die Bauweise bzw. die innere Struktur, grün auf den Zweck bzw. die Funktion und die Außensicht.

Abbildung 9: Grafik Gestaltungskreis



Die Abbildungen 8 und 9 sind Zieldimensionen informatischer Bildung im Elementar- und Primarbereich (Bergner u. a., 2018) entnommen.

## 6.2 Entwicklung des Fächerkanons

Änderungen im Bildungskanon finden nicht häufig statt. Die Elemente der allgemeinen Bildung, die zu dem gegenwärtigen Bildungskanon in Form der Elemente in den Stundentafeln der allgemeinbildenden Schulen geführt haben, wurden vor ca. 150 Jahren identifiziert und grundsätzlich verankert.

Dabei ist maßgeblich, dass der gesellschaftliche Fortschritt an die allgemeine Bildung gekoppelt wird.<sup>1</sup> In einem 50 Jahre dauernden Prozess wird – ökonomischen Erfordernissen Rechnung tragend – gesellschaftlich durchgesetzt, dass die Naturwissenschaften als allgemeinbildend ausgewiesen und als Pflichtbestandteil in die Bildungspläne übernommen wurden. BAECKER geht in der Analyse einige Erkenntnisschritte weiter (vgl. Baecker, 2007, S. 14):

»Die Bedeutung des Computers ist erst dann zu verstehen, wenn man seine Einführung mit der Einführung der Schrift vor über 5000 Jahren<sup>2</sup> und des Buchdrucks vor 500 Jahren vergleicht. Jedes Mal hat sich die Form der Gesellschaft tief greifend verändert. Und jedes Mal hat man erst Jahrhunderte später begriffen, was sich abgespielt hat«

<sup>1</sup> Ergebnis der Aufklärung, die ihrerseits eine gesellschaftlich-philosophische Reaktion auf den Buchdruck darstellt: Jede[r] soll selbst so kritikfähig sein, dass eine Beurteilung von Druckerzeugnissen möglich ist.

<sup>2</sup> Baecker gibt hier »vor 3000 Jahren« an – dabei handelt es sich definitiv um einen Fehler.

<b>Schrift</b>	→	<b>Schule für wenige</b>
<b>Buchdruck</b>	→	<b>Schule für alle (Aufklärung)</b>
<b>Informatik</b>	→	<b>Informatik für alle (informatische Aufklärung)</b>

### 6.3 Entwicklung der Schulinformatik

2019 – 50 Jahre Informatik an NRW-Schulen

Seit 1969 wird Informatik in NRW als Schulfach angeboten.

- Oberstufe
  - Realschule ab Kl. 7
  - Grund- und Leistungskurse Informatik
  - Gesamtschule, Sekundarschule ab Kl. 6/7
- Sekundarstufe I
  - Wahlpflichtbereich Gymnasium ab Kl. 8
  - Grundschule – ab 2015 Projekt »Informatik an Grundschulen (IaG)«

2019 werden wir **50 Jahre Informatik als Schulfach in NRW feiern.**

#### Internationaler Stand – Informatik im Primarbereich

International gibt es in einigen Ländern – vor allem im osteuropäischen Raum und Asien/Ozeanien – ein durchgängiges Pflichtfach Informatik – zumindest ab der Sekundarstufe I – schon seit längerem. Aktuell ziehen viele Nationen der westlichen Welt nach, u. a. Großbritannien oder die Schweiz. Dabei wird überall auch zunehmend der Primarbereich abgedeckt. In Großbritannien gibt es das durchgängige Fach Computing ab der ersten Klasse seit 2014. Im Kern betrachtet es das altersgerechte Erschließen von Informatiksystemen inklusive einfacher Schritte der Programmierung. Die Einführung des neuen Schulfachs wurde von einer groß angelegten Qualifizierungsinitiative der Lehrkräfte begleitet.

In den USA beschreiben die CSTA-Standards für informatische Bildung ab der Primarstufe. Hier wird Zusammenarbeit, Computational Thinking, Nutzungskompetenzen und – weniger als der britische Ansatz – altersgerechtes Programmieren den Mittelpunkt gestellt. Der Schweizer Lehrplan 21 definiert das Fach Informatik und digitale Medien ab der dritten Klasse mit den Themen Daten, Algorithmen und Programmieren. Ein ausführlicher Vergleich der aktuellen internationalen Initiativen und ihrer Ziele und Vorschläge, wie diese nach Deutschland transferiert werden können, ist in *Zieldimensionen informatischer Bildung im Elementar- und Primarbereich* (Bergner u. a., 2018) zu finden.

### 6.4 Gender Gap in der Informatik

Datenquelle: »Das Schulwesen in Nordrhein-Westfalen aus quantitativer Sicht 2016/17 – statistische Übersicht 391« (MSW-NW, 2017)

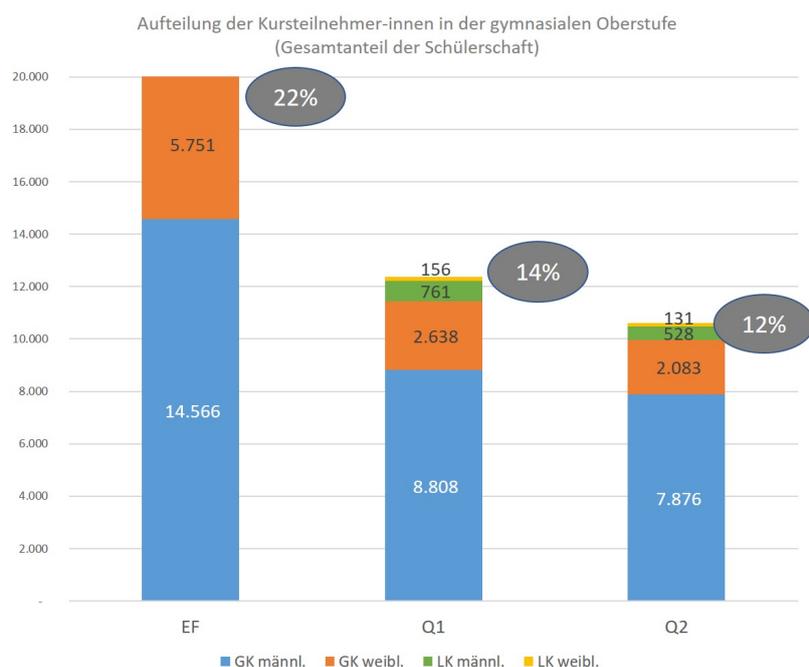


Abbildung 10: Informatik in der Schule – Jungen und Mädchen

- In NRW gibt es das Schulfach Informatik ausschließlich im Wahl- bzw. Wahlpflichtbereich.
- In der Sekundarstufe I belegen über alle Schularten in den Klassenstufen 5–10 je nach Klassenstufe zwischen 6 und 16% aller Schülerinnen und Schüler das Fach Informatik.
- In der gymnasialen Oberstufe belegen unter 20% der Schülerinnen und Schüler einen Informatikkurs. Schülerinnen machen in dieser Gruppe lediglich ein Viertel aus.
- Auffällig ist, dass der Anteil der Schülerinnen und Schüler relativ zur Mathematik von der EF (22%) zur Q2 auf knapp über 12% fällt. Dabei fällt der Anteil der Schülerinnen stärker als der von Schülern von 6,2% in der EF auf 2,4% in der Q2.
- Besonders klein ist der Anteil von Schülerinnen und Schülern, die einen LK belegen (insgesamt nur 0,76% der Schülerinnen und Schüler).

Die frühe Einführung informatischer Bildung könnte sowohl dem schwachen Wahlverhalten insgesamt als auch der Ungleichverteilung zwischen Mädchen und Jungen entgegenwirken.

! **Gendermainstreaming** bezogen auf Informatik kann dadurch eingelöst werden, dass alle **Schüler\*innen** die Chance erhalten, ihre informatische Bildung zu entwickeln.

## 7 Zusammenfassung wesentlicher Argumente für Informatik an Grundschulen

- Insbesondere bei der Wissenschaft Informatik, die einer fundamentalen Veränderung fast aller gesellschaftlichen und ökonomischen Bedingungen die Grundlage liefert, ist es unabdingbar, dass informatisch qualifizierte Lehrkräfte in der Schule eine fachdidaktisch-pädagogisch verantwortliche Gestaltung der informatischen Grundlagen leisten und sie mit aktuellen Entwicklungen untersetzen können.
- Die Idee, dass Schülerinnen und Schüler allein durch die Nutzung von Informatiksystemen im Unterricht quasi *nebenbei Informatik verstehen*, ist wenig zielführend (man lernt auch nicht durch die »Bedienung des Taschenrechners« Mathematik oder durch »Fahrradfahren« Physik oder durch »Haarewaschen« Chemie). Es ist klar, dass nur explizit der informatischen Bildung zuzuordnender Unterricht die Schülerinnen und Schüler in die Lage versetzt, die Grundlagen der Informatik zu verstehen, die unabhängig von aktuellen Informatiksystemen auch in Zukunft gültig sind.

Im Projekt »Informatik an Grundschulen (IaG)«, das von drei Universitäten mit Unterstützung des MSB in NRW durchgeführt wird (MSB NRW, 2017), zeigt sich deutlich, dass die Lehrkräfte hinsichtlich fachlicher und fachdidaktischer Elemente der Informatik qualifiziert werden müssen.

Darüber hinaus wurde deutlich, dass die Ideen der Informatik sich auch **ohne den Einsatz von Informatiksystemen** so entwickeln und umsetzen lassen, dass Informatik durch die Schülerinnen und Schüler spannend erlebt wird.

### Informatik und Medienbildung

Es sieht so aus, als ob es einen Widerspruch oder eine Konkurrenz zwischen den beiden Elementen gibt. Daher ist hier die Stelle, um zu betonen: Es gibt keinen Widerspruch von Seiten der Informatik zur Notwendigkeit der Medienbildung!

Allerdings gibt es – aus Sicht der Informatikfachdidaktik – Klarheit darüber, dass Informatik für Medienbildung als Voraussetzung notwendig ist. Das bedeutet in der Konsequenz, dass die Lehrkräfte als Vermittlungskompetenz für die Medienbildung notwendig informatische Kompetenzen benötigen.

! Nur mit informatischen Grundlagen kann der Gestaltungsanspruch der Medienbildung – als wesentlicher Baustein der allgemeinen Bildung – auf einer fachlichen Grundlage entwickelt werden.

### Informatik mit Informatiksystemen vermitteln

In den drei Modulen des Projektes Informatik an Grundschulen werden informatische Grundlagen und -prinzipien ohne den Einsatz von Informatiksystemen wie einem Laptop oder Tablet vermittelt. Als Ergänzung dazu ist eine Vertiefung der erworbenen Kompetenzen mit dem Einsatz von Informatiksystemen wünschenswert. So kann beispielsweise die Robotersteuerung, die im Modul »Wie funktioniert der Roboter« mittels Papierbefehlen eingeführt wird, mit Informatiksystemunterstützung umgesetzt werden. Dazu stehen zum einem Werkzeuge wie Scratch oder Scratch Junior zur Verfügung, die eine grafische Programmierung im Browser bzw. als App ermöglichen. Darüber hinaus gibt es auch Informatiksysteme, die speziell für die Zielgruppe der Grundschul Kinder entwickelt wurde.

Hier könnten zum Beispiel der Holzroboter Cubetto, die Spielzeugroboter Dash & Dot, der Selbstbausatz Lego Wedo und die Calliope Mini eine sinnvolle Ergänzung darstellen (vgl. Abbildungen 11 und 12)). Neben Robotiksystemen gibt es noch weitere schülergerechte Informatiksysteme als Lernumgebungen, welche die Ergebnisse der Programmierung und Aspekte der Sensorsteuerung in die reale Welt der Kinder überträgt.

Aktuell ist die Calliope Mini als programmierbares Board in ganz Deutschland im Gespräch – hier wird der Ansatz gewählt, dass ohne eigene Programmierung 25 vorinstallierte Szenarien genutzt werden können.

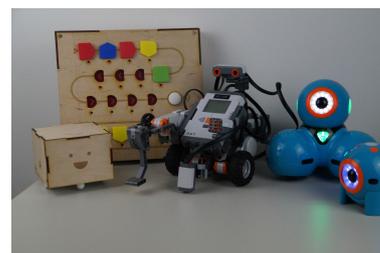


Abbildung 11: Informatiksysteme als Lernwerkzeuge

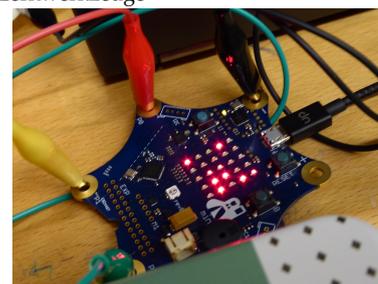


Abbildung 12: Calliope mini

## Literatur

- Baecker, Dirk. *Studien zur nächsten Gesellschaft*. suhrkamp taschenbuch wissenschaft 1856. Frankfurt a. M.: Suhrkamp Verlag, Okt. 2007. ISBN: 3-518-29456-3. URL: <https://t1p.de/hd9q> (besucht am 13. 03. 2019).
- Balzert, Helmut. *Informatik: 1. Vom Problem zum Programm – Hauptband*. 1. Aufl. München: Hueber-Holzmann Verlag, 1976.
- Bergner, Nadine u. a. *Zieldimensionen informatischer Bildung im Elementar- und Primarbereich*. Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich. Berlin: Stiftung »Haus der kleinen Forscher«, 2018.
- Computer Science Teachers Association (CSTA). *K-12 Computer Science Standards, Revised 2017*. 27. Aug. 2018. URL: <https://t1p.de/hzuu> (besucht am 13. 03. 2019).
- Fricke, Martin, Kathrin Haselmeier u. a. *Informatik im Unterricht – So geht's*. Plakatgrafik zu Informatik an Grundschulen. Wuppertal, 2016. URL: <https://uni-w.de/op> (besucht am 07. 03. 2019).
- Fricke, Martin und Felix Schaumburg. »Der Medienkompetenzrahmen in Nordrhein-Westfalen«. In: *LOG IN* 189/190.189/190 (2018). Hrsg. von Ludger Humbert und Bernhard Koerber, S. 32–37. ISSN: 0720-8642. URL: <http://uni-w.de/lap> (besucht am 28. 01. 2019).
- Gesellschaft für Informatik e. V., Hrsg. *Kompetenzen für informatische Bildung im Primarbereich. Empfehlungen der Gesellschaft für Informatik e. V.* Erarbeitet vom Arbeitskreis »Bildungsstandards Primarbereich« – Beschluss des GI-Präsidiums vom 31. Januar 2019 – wird in gedruckter Form der *LOG IN* 39 (2019) Heft 191/192 beigelegt. 7. Feb. 2019. URL: <http://uni-w.de/lgm> (besucht am 09. 03. 2019).
- Humbert, Ludger. *Didaktik der Informatik – mit praxiserprobtem Unterrichtsmaterial*. 2., überarbeitete und erweiterte Aufl. Leitfäden der Informatik. Wiesbaden: B.G. Teubner Verlag, Aug. 2006. ISBN: 3-8351-0112-9.
- *Veranstaltungskarte – Informatik im Alltag – Wintersemester 2018/2019 – Bergische Universität Wuppertal*. Okt. 2018. URL: <http://uni-w.de/lch> (besucht am 09. 03. 2019).
- Klafki, Wolfgang. *Neue Studien zur Bildungstheorie und Didaktik: Zeitgemäße Allgemeinbildung und kritisch-konstruktive Didaktik*. 3. Aufl. Weinheim, Basel: Beltz Verlag, 1993.
- KMK, Hrsg. *Strategie der Kultusministerkonferenz »Bildung in der digitalen Welt«*. Beschluss der Kultusministerkonferenz vom 08.12.2016. Berlin, 21. Dez. 2016. URL: <https://t1p.de/hm0j> (besucht am 22. 02. 2019).
- MSB NRW, Hrsg. *Informatik an Grundschulen – Ziele. Pilotprojekt zur Erprobung von Konzepten zur informatischen Bildung im Rahmen des Sachunterrichts an Grundschulen*. MSB NRW – Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen. Juli 2017. URL: <https://t1p.de/pods> (besucht am 25. 01. 2019).
- MSW-NW. *Das Schulwesen in Nordrhein-Westfalen aus quantitativer Sicht 2016/17 – statistische Übersicht 391*. MSW-NW – Ministerium für Schule und Weiterbildung Nordrhein-Westfalen. Düsseldorf, 3. Mai 2017. URL: <https://t1p.de/g7cr> (besucht am 13. 03. 2019).
- Thomas, Marco. *Didaktik der Informatik II – Vorlesungsmaterial – Universität Dortmund – Fachbereich Informatik Wintersemester 2002/2003*. Dez. 2002.

Abbildungsnachweise

Kurzbezeichnung – Autor – Lizenz .....	Seite
Projektlogo IaG – Autor: Bergische Universität Wuppertal – 	GB 01
Logo des MSB – Autor: Ministerium für Schule und Bildung Nordrhein-Westfalen (MSB) – unfrei	GB 02
Mikrowelle, Aufzug, Spielautomat, EEG, Diebstahlstele – Autor: Johannes Magenheim – 	GB 03
TV, Ampel, Supermarktkasse – Autor: Johannes Magenheim – 	GB 04
Daten und Information – Autor: Ludger Humbert – 	GB 05
Informatischer Modellierungskreis – Autor: Ludger Humbert – 	GB 06
Fachgebiete der Informatik: Kerninformatik und mehr – Autor: Marco Thomas – 	GB 06
Fachgebiete der Informatik, Modulzuordnung – Autor: Ludger Humbert – 	GB 07
Informatik an Grundschulen – Kompetenzen – Autor: Martin Fricke u. a. – 	GB 08
Grafik Explorationskreis – Autor: Nadine Bergner u. a. – 	GB 09
Grafik Gestaltungskreis – Autor: Nadine Bergner u. a. – 	GB 10
Informatik in der Schule – Jungen und Mädchen – Autor: Ulrik Schroeder – 	GB 11
Informatiksysteme als Lernwerkzeuge – Autor: Nadine Bergner – 	GB 13
Calliope mini – Autor: Dorothee Müller – 	GB 13



[Verweis auf Lehrerhandreichungen und Materialsammlungen zum Einsatz im Unterricht]





# INFORMATIK AN GRUNDSCHULEN

»Digitale Welt«

## Lehrerhandreichung zum Modul

**RWTH Aachen**  
**Lehr- und Forschungsgebiet Informatik 9 – Lerntechnologien**



Ministerium für  
Schule und Bildung  
des Landes Nordrhein-Westfalen



# Digitale Welt



Entstanden durch  
Nadine Bergner, Dieter Frohnhofen, Stefan Moritz

Stand: März 2019

## Inhaltsverzeichnis

Das Projekt „Informatik an Grundschulen“.....	DW 03
Übersicht über Modul 1: Digitale Welt.....	DW 03
Vorwort: Worum geht es?.....	DW 03
Einführung (UE0).....	DW 03
Erste Unterrichtseinheit (UE1).....	DW 04
Zweite Unterrichtseinheit (UE2).....	DW 04
Dritte Unterrichtseinheit (UE3).....	DW 04
Kompetenzen.....	DW 04
Modellierungskreislauf der Informatik.....	DW 07
Situation (1).....	DW 07
Modell (2).....	DW 07
Konsequenz (3).....	DW 08
Ergebnisse (4).....	DW 08
Übersicht über die Unterrichtseinheiten von Modul 1.....	DW 09
UE0: Einführung.....	DW 10
Lernziel der Stunde.....	DW 11
Inhalt der Stunde.....	DW 11
Dauer.....	DW 11
Materialien.....	DW 11
Fachliche Analyse zur UE0: Einführung.....	DW 12
Zusatzinformation.....	DW 14
Empfohlene Vorgehensweise für UE0: Einführung.....	DW 15
UE1: Binärdarstellung.....	DW 16
Lernziel der Stunde.....	DW 17
Inhalt der Stunde.....	DW 17
Dauer.....	DW 17
Materialien.....	DW 17
Fachliche Analyse zur UE1: Binärdarstellung.....	DW 18
Zusatzinformation.....	DW 19
Negative Zahlen, Fließkommazahlen und Buchstaben.....	DW 20
Bits & Bytes.....	DW 21
Weitere Zahlensysteme.....	DW 22
Empfohlene Vorgehensweise für UE1: Binärdarstellung.....	DW 23
UE2: Datenübertragung.....	DW 25
Lernziel der Stunde.....	DW 26
Inhalt der Stunde.....	DW 26
Dauer.....	DW 26
Materialien.....	DW 26
Fachliche Analyse zur UE2: Datenübertragung.....	DW 27
Zusatzinformation.....	DW 28
Empfohlene Vorgehensweise zur UE2: Datenübertragung.....	DW 32
UE3: Fehlererkennung.....	DW 34
Lernziel der Stunde.....	DW 35
Inhalt der Stunde.....	DW 35
Dauer.....	DW 35
Materialien.....	DW 35
Fachliche Analyse zur UE3: Fehlererkennung.....	DW 36
Zusatzinformation.....	DW 37
Empfohlene Vorgehensweise für UE3: Fehlererkennung.....	DW 39
Abbildungsverzeichnis.....	DW 41

## Das Projekt „Informatik an Grundschulen“

In dem Projekt „Informatik an Grundschulen“ beleuchten drei Module unterschiedliche fachlich-methodische Schwerpunkte der Informatik. Die Auswahl der Themen erfolgte unter fachdidaktischen, lernpsychologischen und allgemeinbildenden Gesichtspunkten. Für die unterrichtliche Umsetzung in der Schule ist jeweils ein zeitlicher Rahmen von ca. 6 bis 8 Stunden vorgesehen. Grundlegende fachliche und fachdidaktische Hinweise zum Projekt „Informatik an Grundschulen“ sind im *Einführungsband* nachzulesen. Zu jedem Modul gibt es einen spezifischen Modulband, welcher aus Lehrerbegleitheft (dieses halten Sie für das Modul „Digitale Welt“ gerade in Händen bzw. sehen es auf dem Monitor) und Lernmaterialien für Schülerinnen und Schüler, die teilweise bereits konkrete Kopiervorlagen beinhalten.

Alle drei Module folgen einem kompetenzbasierten fachdidaktischen Konzept, das Kompetenzen als Einheit von kognitiven (fachlichen) und nicht-kognitiven (motivationalen, volitionalen) Komponenten versteht. In dem Projekt „Informatik an Grundschulen“ werden die mit dem Informatikunterricht verbundenen Kompetenzerwartungen mit einem Modell beschrieben, das sich an den Empfehlungen der Gesellschaft für Informatik (GI) zur Informatik an der Grundschule („Empfehlungen für Bildungsstandards“) und einer Expertise des „Haus der kleinen Forscher“ (HdkF) zur frühen informatischen Bildung orientiert. Eine genauere Darstellung der Konzepte und zur fachdidaktischen Einordnung der Module befindet sich im Einführungsband zum Projekt.

Entsprechend dieser Empfehlungen werden auch im Projekt „Informatik an Grundschulen“ die mit den Modulen verbundenen Kompetenzerwartungen *Kompetenzfeldern* zugeordnet, die sich jeweils durch eine *Prozess-* und eine *Inhaltskomponente* beschreiben lassen. Danach werden beobachtbare Handlungen der Schülerinnen und Schüler (Prozesse) einem fachlichen Themenbereich zugeordnet (Inhaltsbereich). Eine detaillierte Darstellung des Konzepts findet sich im Einführungsband.

## Übersicht über Modul 1: Digitale Welt

### Vorwort: Worum geht es?

Information ist neben Stoff und Energie die dritte Grundgröße der Natur (vgl. Wiener 1961). Der Umgang mit Information führt zu einem permanenten gesellschaftlichen Wandel und sollte bereits von Grundschülerinnen und Grundschülern kennengelernt und in Grundzügen verstanden werden. So ist das Ziel des Moduls 1, welches im Rahmen des NRW-weiten Projektes „Informatik an Grundschulen“ am Standort Aachen entwickelt wurde, Kindern der Grundschule die Möglichkeit zu eröffnen, Primärerfahrungen im Bereich der digitalen Welt zu sammeln.

In den vier entwickelten Unterrichtseinheiten sollen Alltagsfragen der Kinder im Bereich der Datenspeicherung und -übertragung aufgegriffen und erläutert, aber auch abstrakte Alltagsprobleme, wie z.B. Fehler bei der Datenübertragung und die Möglichkeit ihrer Erkennung in einfachen Szenarien erarbeitet und dadurch für Kinder erlebbar gemacht werden. Als Grundlage der digitalen Datenübertragung bildet die Binärdarstellung von digitalen Daten in 0 und 1 das verbindende Element des Moduls.

Ziel der Unterrichtsreihe ist es, Kindern einen Einblick in die *Digitale Welt* zu ermöglichen. Die Unterrichtsreihe ist neben einer **45-minütigen Einführungsstunde**, die an die Erfahrungen der Kinder anknüpft, auf **drei 90-minütige Unterrichtseinheiten** ausgelegt.

### Einführung (UE0)

Ausgangspunkt des Moduls „Digitale Welt“ sind *Codes*, also bestimmte Darstellungsformen von Daten. Spezifisch werden hier QR-Codes behandelt, da davon ausgegangen wird, dass diese der Mehr-

heit der Kinder bereits im Alltag (z. B. auf Plakaten, Postsendungen oder Bahntickets) begegnet sind. Entscheidend für dieses Modul ist dabei die Darstellung in digitaler Form, also über Anordnung der schwarzen und weißen kleinen Quadrate in dem Gesamtbild des QR-Codes.

### Erste Unterrichtseinheit (UE1)

Anknüpfend an die *Datenrepräsentation* in einem QR-Code werden in dieser Einheit *Binärzahlen* eingeführt, die Daten ebenfalls über zwei Zustände (beispielsweise rot und blau oder auch 0 und 1) darstellen. Hierbei wird der Bezug zum Informatiksystem (Computer, Tablet, Smartphone, etc.) hergestellt, da alle diese Systeme mittels der beiden Zustände 0 für „Strom aus“ und 1 für „Strom an“ arbeiten. Die Kinder lernen also wie Daten in Informatiksystemen dargestellt werden, indem sie erst Dezimalzahlen und später auch Buchstaben in ihre Binärdarstellung umwandeln und umgekehrt. Es erfolgt die Erkenntnis, dass im vierstelligen Binärcode alle Zahlen von 0-15 darstellbar sind und für ein größeres Zahlenspektrum eine fünfte Stelle hinzugenommen werden muss.

### Zweite Unterrichtseinheit (UE2)

Nachdem die Darstellung von Daten im Computer (Laptop, Tablet, Smartphone) verinnerlicht wurde, widmet sich diese zweite Einheit der *Übertragung* dieser Daten (beispielsweise über das Internet). Dabei lernen die Kinder das grundlegende Prinzip der Eingabe-Verarbeitung-Ausgabe (EVA-Prinzip) der Informatik kennen. Auch ahmen sie die Funktionsweise eines Informatiksystems nach, indem sie selbst Botschaften binär codieren und diese dann an Mitschülerinnen und Mitschüler übertragen, die die Botschaft ihrerseits dann entsprechend decodieren und lesen. An dieser Stelle wird das Wissen über die Binärdarstellung von Dezimalzahlen auf die Darstellung von Buchstaben transferiert. Die Kinder sind somit in der Lage, mit dem angeeigneten Wissen zum Binärcode einfache Botschaften zu übertragen und auch empfangene Botschaften zu decodieren.

### Dritte Unterrichtseinheit (UE3)

Die abschließende Einheit thematisiert den Umgang mit möglichen Fehlern, die bei der Eingabe oder auch Übertragung von Daten auftreten können. Die Kernbotschaft dabei liegt in der Tatsache, dass es Fehlererkennungsmechanismen gibt, die hier in sehr einfacher Form an dem konkreten Beispiel eines Prüfbits (eine zusätzliche Ziffer, welche übertragen wird, um die Korrektheit einer Nachricht zu gewährleisten) erkundet werden können. Die Kinder erfahren, warum Fehlererkennung ein wichtiges Thema (z. B. bei der Steuerung großer Maschinen) darstellt und erlangen eine Grundvorstellung darüber, wie solche Algorithmen arbeiten.

## Kompetenzen

Über die vier Einheiten hinweg werden folgende Kompetenzen bei den Kindern erzielt.

Die Schülerinnen und Schüler

- (DW – K1)** beschreiben ihren persönlichen Umgang mit Informatiksystemen,
- (DW – K2)** erläutern das Prinzip der Dateneingabe und der Datenübertragung,
- (DW – K3)** begründen die Einschränkung auf die Zustände *Strom an* und *Strom aus*,
- (DW – K4)** erklären, wie Information in digitale Daten umgewandelt werden kann,
- (DW – K5)** führen die Umwandlung von Dezimalzahlen und Buchstaben in Binärzahlen wie auch in umgekehrter Richtung selbst durch,
- (DW – K6)** erläutern die Notwendigkeit von Fehlererkennungsmechanismen bei der Datenübertragung und
- (DW – K7)** beschreiben ein vorgegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle und wenden dies an.

Für das Modul „Digitale Welt“ erläutert die unten stehende Tabelle in Form einer *Heatmap* die entsprechenden Zuordnungen der Kompetenzen zu den Inhalts- und Prozessbereichen. Dabei wird deutlich, dass sich die in diesem Lehrerbegleitheft später genauer erläuterten und einzelnen Unterrichtsstunden zugeordneten Kompetenzerwartungen DW-K1 bis DW-K7 nicht eindeutig einem Kompetenzfeld zuordnen lassen, sondern oft mehrere von ihnen adressieren. Mit der hier abgebildeten *Heatmap* soll daher verdeutlicht werden, welche Kompetenzfelder in dem Modul schwerpunktmäßig angesprochen werden (dunkles Grün), welche ebenfalls von Bedeutung sind (helles Grün) und welche in diesem Modul lediglich ergänzende Funktionen besitzen (weiß). Wie sich die Erwartungen mit dem in diesem Modul beschriebenen Informatikunterricht letztlich praktisch umsetzen lassen und welcher Schwerpunkt hauptsächlich fokussiert wird, liegt in der gestalterischen Freiheit der einzelnen Lehrkraft. Die *Heatmap* soll hier nur einen Orientierungsrahmen liefern.

Inhaltsbereiche Prozessbereiche	(I1) Information & Daten	(I2) Algorithmen & Programmierung	(I3) Sprachen & Automation	(I4) Informatiksysteme	(I5) Informatik, Mensch & Gesellschaft
(P0) Interagieren & Explorieren				beschreiben ihren persönlichen Umgang mit Informatiksystemen <b>(DW - K1)</b>	beschreiben ihren persönlichen Umgang mit Informatiksystemen <b>(DW - K1)</b>
(P1) Modellieren & Implementieren		führen die Umwandlung von Dezimalzahlen und Buchstaben in Binärzahlen wie auch in umgekehrter Richtung selbst durch <b>(DW - K5)</b>			
(P2) Begründen & Bewerten	begründen die Einschränkung auf die Zustände <i>Strom an</i> und <i>Strom aus</i> <b>(DW - K3)</b>	erläutern die Notwendigkeit von Fehlererkennungsmechanismen bei der Datenübertragung <b>(DW - K6)</b>		begründen die Einschränkung auf die Zustände <i>Strom an</i> und <i>Strom aus</i> <b>(DW - K3)</b>	

<p>(P3) Strukturieren &amp; Vernetzen</p>	<p>erläutern das Prinzip der Dateneingabe und der Datenübertragung <b>(DW - K2)</b></p>				<p>erklären, wie Information in digitale Daten umgewandelt werden kann <b>(DW - K4)</b></p>
<p>(P4) Kommunizieren &amp; Kooperieren</p>		<p>beschreiben ein vorgegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle und wenden dies an <b>(DW - K7)</b></p>		<p>beschreiben ihren persönlichen Umgang mit Informatiksystemen <b>(DW - K1)</b></p>	
<p>(P5) Darstellen &amp; Interpretieren</p>					

## Modellierungskreislauf der Informatik

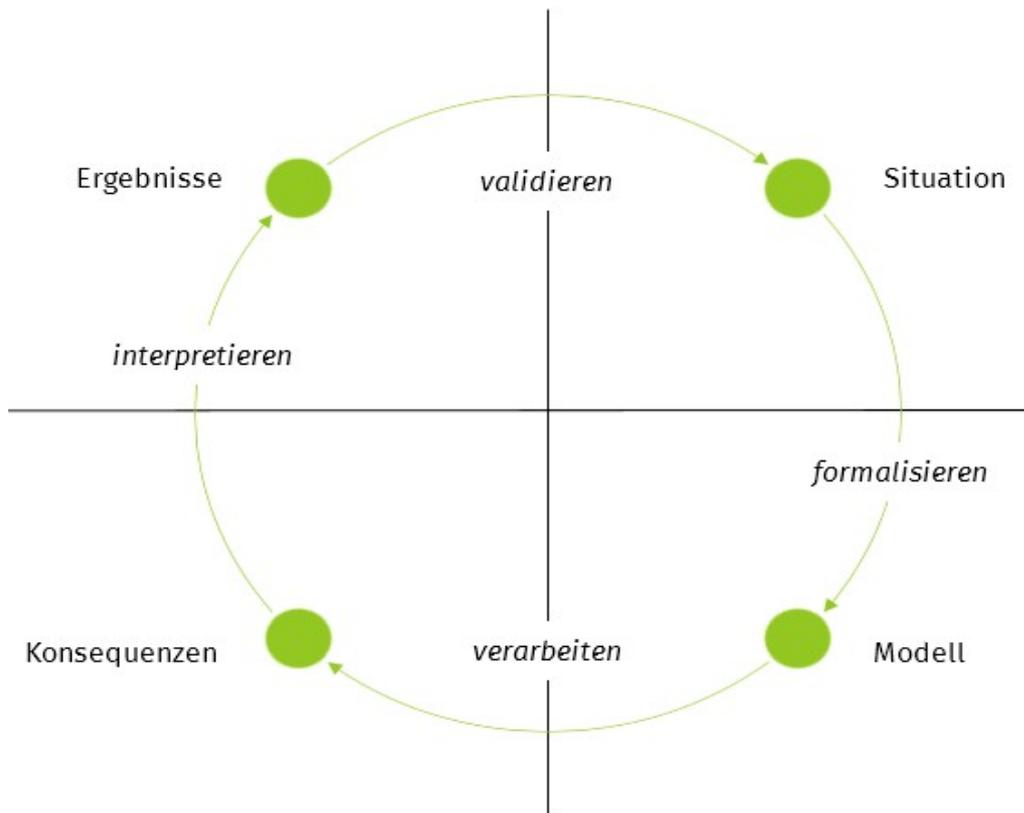


Abbildung 1: Modellierungskreislauf der Informatik (nach Grafik von Ludger Humbert)

Im informatischen Modellierungskreislauf (vgl. Abb. 1) lernen die Schülerinnen und Schüler, wie eine Problemstellung der Alltagswirklichkeit aus Sicht der Informatik bearbeitet wird.

Der Kreislauf unterscheidet horizontal die Ebenen *Welt* und *Informatik*. Vertikal werden die Ebenen *Problem* und *Lösung* unterschieden. Hieraus ergeben sich die vier Kreissektoren mit den Zuständen *Situation*, *Modell*, *Konsequenzen* und *Ergebnisse*.

Jeder Übergang von einem Zustand zum nächsten entspricht einem festgelegten Arbeitsschritt.

### Situation (1)

Der Zustand *Situation* stellt in der realen Welt ein Problem dar.

Die Schülerinnen und Schüler sollen Botschaften verschicken. Das (simulierte) Problem dabei ist, dass die Botschaft über ein Kabel mittels Stromimpulsen verschickt werden muss. Also gibt es nur zwei mögliche Signale (Zustände), nämlich Strom an/Strom aus.

### Modell (2)

Um ein Modell für dieses Problem zu entwickeln, muss das Problem formalisiert werden.

Um alle Botschaften verschicken zu können, muss das Alphabet (eventuell mit Satzzeichen) in einen Code übersetzt werden, der nur 2 Zeichen enthält. Diese werden dann den Zuständen Strom an/Strom aus zugeordnet.

### Konsequenz (3)

Aus dem Modell wird eine Konsequenz gezogen.

**Die Buchstaben und Satzzeichen werden in einen fünfstelligen Lämpchencode übersetzt, wobei die Lämpchen leuchten (gelb) oder nicht leuchten (weiß).**

### Ergebnisse (4)

Die Konsequenz wird nun für die reale Welt interpretiert und die Ergebnisse bilden eine Lösung für das Problem.

Die Schülerinnen und Schüler können anhand einer Codetabelle und anhand mit Lämpchen bestückter Endlosstreifen (zur Übersichtlichkeit sind immer 5 Lämpchen durch eine Markierung getrennt) sowie einem gelben Stift beliebige Botschaften verfassen und verschicken. Der Empfänger kann dann mittels einer weiteren Codetabelle die Botschaft decodieren.

Auch eine tatsächliche Übertragung der Botschaften mittels Lämpchen, Schaltern und Kabeln (fünfpolig) ist einfach realisierbar und somit das Problem nicht länger nur simuliert.

## Übersicht über die Unterrichtseinheiten von Modul 1

Seite	Unterrichtseinheit	Thema	Lernziele / erlernte Kompetenzen	Dauer	
10		<b>Einführung</b>	QR-Code; Zwei Zustände – schwarze und weiße Kästchen	Die Kinder werden mittels QR-Code an das Thema <i>Digitale Welt</i> herangeführt.	45 min
16		<b>Binärdarstellung</b>	Einschränkung von Information auf zwei Farben; technische Umsetzung durch die elektrischen Zustände AN/AUS; lesen von vierstelligen Binärcodes	Die Kinder <ul style="list-style-type: none"> <li>• begründen die Einschränkung auf die beiden Zustände <i>Strom an</i> (entspricht 1) und <i>Strom aus</i> (entspricht 0).</li> <li>• erklären, wie Daten in Binärdarstellung umgewandelt werden.</li> <li>• führen die Umwandlung von Dezimalzahlen in Binärzahlen, wie auch in umgekehrter Richtung, selbst durch.</li> </ul>	90 min
25		<b>Datenübertragung</b>	Darstellung von Buchstaben als fünfstellige Binärcodes; Verfassen von Texten in digitaler Form; Verschicken von Texten mittels „elektrischer Datenleitung“	Die Kinder <ul style="list-style-type: none"> <li>• erläutern das Prinzip der Eingabe, Verarbeitung und Ausgabe von Daten als grundlegendes Arbeitsprinzip von Informatiksystemen.</li> <li>• führen die Umwandlung von Buchstaben in Binärzahlen, wie auch in umgekehrter Richtung, selbst durch.</li> </ul>	90 min
34		<b>Fehlererkennung</b>	Fehler beim Übertragen von digitalen Daten; Erkennung von Fehlern durch zusätzliche Überprüfungsdaten als Instrument (Prüfbits)	Die Kinder <ul style="list-style-type: none"> <li>• erläutern das Prinzip der Eingabe, Verarbeitung und Ausgabe von Daten als grundlegendes Arbeitsprinzip von Informatiksystemen.</li> <li>• erläutern die Notwendigkeit von Fehlererkennungsmechanismen bei der Datenübertragung.</li> <li>• beschreiben ein vorgegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle mit eigenen Worten.</li> </ul>	90 min



# UE0: Einführung



## Lernziel der Stunde

Die Kinder kennen QR-Codes<sup>1</sup> und verstehen die Funktionsweise solcher Codes. Sie erkennen, dass QR-Codes von zwei verschiedenen Zuständen abhängen.

Die Schülerinnen und Schüler

**(DW – K1)** beschreiben ihren persönlichen Umgang mit Informatiksystemen.

## Inhalt der Stunde

Die Kinder werden mittels QR-Code an das Thema *Digitale Welt* herangeführt.

Dabei sollen die Schülerinnen und Schüler aus ihrer eigenen Welt QR-Codes wiedererkennen. Mit Hilfe der Kästchen in den Farben Schwarz und Weiß wird den Kindern verdeutlicht, dass QR-Codes aus zahlreichen Quadraten in zwei verschiedenen Zuständen (schwarz oder weiß) bestehen.

Mit Hilfe von bestimmten Scannern können QR-Codes ausgelesen werden und die Schülerinnen und Schüler sehen, was sich hinter den QR-Codes verbirgt. Ein solcher Scan kann mittels (kostenfreier) Smartphone-Apps von der Lehrkraft bzw. den Kindern selbst live im Klassenzimmer durchgeführt werden. Ist ein solcher Scanner nicht vorhanden, können die Lösungskarten benutzt werden.

Außerdem wird dargelegt, dass Information zum Beispiel mittels QR-Code codiert werden kann. Um sie anschließend zu decodieren, braucht man beispielsweise einen Scanner.

## Dauer

ca. 45 Minuten

## Materialien

- QR-Code-Scanner (z.B. in Form einer Smartphone-App)
- Kopiervorlagen:
  - UE0.1
  - UE0.2
  - UE0.3 a-p
  - UE0.4
  - eventuell UE0.5
- Schwarze Filzstifte

---

<sup>1</sup> Die Bezeichnung QR-Code ist eingetragenes Warenzeichen der Firma *Denso Wave Incorporated*; die Codes an sich können frei verwendet werden.



## Fachliche Analyse zur UE0: Einführung

Die erste Unterrichtseinheit handelt von QR-Codes (Quick Response, „schnelle Antwort“). QR-Codes sind mittlerweile sehr verbreitet und finden in verschiedenen Bereichen eine Anwendung. Die wohl bekannteste Anwendung ist in Werbungen. So werden QR-Codes auf Plakaten, Flyern und auch Visitenkarten gedruckt, um den Inhalt dieser Werbung oder auch einer Webseite direkt aufrufen zu können. QR-Codes sind aber auch in ganz anderen Bereichen des Lebens vorzufinden. So werden mit Hilfe von QR-Codes digitale Briefmarken genutzt, oder auch das Ticket der Bahn kann sowohl auf dem eigenen Handy, als auch auf Papier als QR-Code dargestellt werden. Da diese kleinen quadratischen Muster vielfach Anwendung im Alltag aller Menschen finden, ist es umso wichtiger, diese zu verstehen.

Der QR-Code ist ein zweidimensionaler Code, das heißt zunächst, dass sich hinter diesem Muster eine Information verbirgt. Dieses Muster hat die Form einer quadratischen Matrix, welche wiederum schwarze und weiße Quadrate enthält. So werden die gewünschten Daten in den jeweiligen schwarzen und weißen Quadraten kodiert. Insgesamt bedeutet dies, dass man zum Kodieren von bestimmten Informationen bzw. Daten nur zwei Farben braucht, in diesem Fall *schwarz* und *weiß*. Genauer genommen handelt es sich hierbei nicht um die Farben, sondern andere Symbole, die das Kodieren übernehmen. Diese Symbole sind in der Informatik bzw. in allen Informatiksystemen die Symbole *0* und *1*. Weiße Quadrate stehen dann für das Symbol *0* und schwarze für *1*. Aus diesem Grund spricht man auch davon, dass die Daten *binär* kodiert sind. Eine ausführliche Beschreibung von Binärzahlen und -darstellungen findet sich in der fachlichen Analyse zur UE1.

Zusammengefasst sind QR-Codes im Prinzip kleine Datenspeicher, die bis zu einer DIN A4 Seite Text fassen können.

Ein QR-Code besteht grundsätzlich aus drei Teilen:

- der Versionsinformation (gelber Bereich),
- dem benutzten Datenformat (blauer Bereich) und
- dem Datenteil (roter Bereich).



Die Versionsinformation ist in der oberen rechten und in der unteren linken Ecke kodiert (gelber Bereich). Heutzutage werden zwischen 40 verschiedenen Versionen unterschieden, jedoch werden beispielsweise für Marketingzwecke die Versionen 1 bis 7 verwendet. Für die Versionen gilt: Je höher die Version ist, desto mehr Quadrate sind im QR-Code enthalten (und desto mehr Informationen können gespeichert werden).

Abbildung 2: QR-Code mit markierten Bereichen

Version 1 enthält 21x21 Quadrate, Version 2 enthält 25x25 Quadrate und die höchste Version, also Version 40 hat bereits 177x177 Quadrate.

Das Datenformat wird in den Ecken des Codes angezeigt (blauer Bereich). Das Format legt fest, wie der Scanner die weiteren Daten interpretieren soll bzw. auch wie groß der Bereich der kodierten Daten ist.

Der größte Teil der Fläche steht den eigentlichen Daten zur Verfügung (roter Bereich). Hier werden die kodierten Daten in einem Muster aus schwarzen und weißen Kästchen dargestellt. Diese werden als Binärzahlen aufgefasst, deren Bedeutung abhängig vom gewählten Dateiformat interpretiert wird. Diese Daten sind redundant, also mehrfach vorhanden, so dass Fehler beim Scannen des QR-Codes korrigiert werden können. So wird ermöglicht, dass ein QR-Code auch dann gescannt werden kann, wenn er teilweise abgedeckt oder beschädigt ist.

Hierfür gibt es insgesamt vier verschiedene Level. *Level L* ist das niedrigste Level und erlaubt es, dass der QR-Code bis 7% verdeckt bzw. beschädigt sein kann. Dann gibt es noch *Level M* (bis zu 15% Fehlerkorrektur), *Level Q* (bis zu 25% Fehlerkorrektur) und *Level H* (bis zu 30% Fehlerkorrektur). Hierbei gilt: Je höher die Fehlerkorrektur ist, desto mehr kann ein QR-Code beschädigt werden/



sein, aber desto größer wird er auch. Ein Beispiel zur Leistungsfähigkeit der Fehlerkorrektur ist in Abbildung 2 dargestellt.

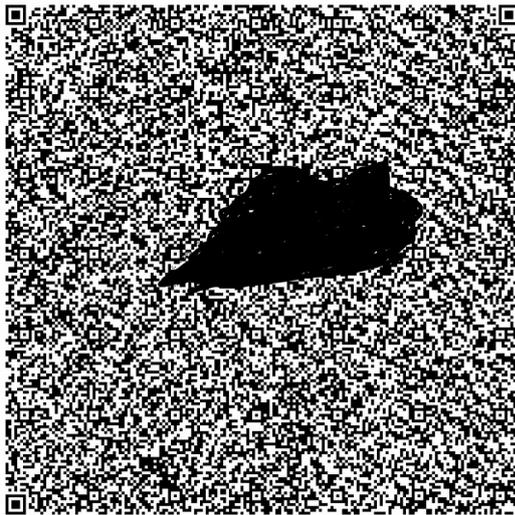


Abbildung 3: QR-Code mit Fehler

Legt man die maximale Größe des Codes von 177x177 Quadraten bei minimalem Level der Fehlerkorrektur (L) zugrunde, so lassen sich 23.648 Bit (2.956 Byte) an binären Informationen speichern. Dies ergibt 7089 Dezimalzahlen bzw. 4296 alphanumerische Zeichen (<http://www.qrcode.com/en/about/version.html>, in Englisch). Zum Vergleich: Auf eine A4-Seite passen bei Schriftgröße 12 mit Rand ca. 4000 Zeichen.

Die drei (in kleineren QR-Codes auch weniger, bzw. in größeren QR-Codes auch mehr) Quadrate in den Ecken dienen zur Ausrichtung des QR-Codes.

Ursprünglich wurde der QR-Code 1994 von dem japanischen Automobilzulieferer *Denso Wave* zu Logistikzwecken entwickelt. Die Spezifikationen des Codes sind frei zugänglich, einzig die Bezeichnung *QR-Code* ist eingetragenes Warenzeichen der Firma *Denso Wave Incorporated* (näheres hierzu siehe <http://www.qrcode.com/en/faq.html#patentH2Title>, in Englisch).

Die QR-Codes zählen zu den zweidimensionalen Codes, da sie, im Gegensatz z. B. zu den bekannten Strichcodes wie etwa GTIN-13, sowohl horizontal als auch vertikal Informationen enthalten. Eine Besonderheit sind die Design-QR-Codes, die sich die Fehlerkorrektur zunutze machen, um Logos, Schriftzüge o. ä. darzustellen. Außer den normalen QR-Codes gibt es noch die Micro-QR-Codes, die deutlich kleiner sind als reguläre QR-Codes und weniger Daten enthalten können. Des Weiteren gibt es den iQR-Code, der eine Weiterentwicklung des normalen QR-Codes darstellt und nicht mehr auf die quadratische Darstellung beschränkt ist. Eine weitere Variante ist der Secure-QR-Code (SQRC), bei dem Lesebeschränkungen eingebaut sind, um sensible Daten zu schützen. Dabei enthalten diese Codes öffentliche und private Daten, wobei spezielle Scanner benötigt werden, um die privaten Daten lesen zu können. Das Erscheinungsbild der SQRC ist dasselbe wie das der regulären QR-Codes. Schließlich gibt es noch die sogenannten Frame-QR-Codes, die in der Mitte des quadratischen Codes einen freien Bereich zur Verfügung stellen, der genutzt werden kann, um z. B. Logos, Authentifizierungsmerkmale o. ä. darzustellen. Einige Beispiele für die verschiedenen Codes sind in den folgenden Abbildungen dargestellt.



Abbildung 4: Frame-QR-Code



Abbildung 5: Design-QR-Code



Abbildung 6: Micro-QR-Code



Abbildung 7: iQR-Code



## Zusatzinformation

Die Generierung eines regulären QR-Codes erfolgt nach dem folgenden Schema: Aus der Länge des zu codierenden Textes und dem gewünschten Grad der Fehlerkorrektur ergibt sich die benötigte Größe des QR-Codes. Als erster Schritt werden die Positionsmarker, die Ausrichtungsmuster sowie die Synchronisationslinien eingebaut. Aus dem zu codierenden Text wird, abhängig vom Zeichensatz (Zahlen, japanische Schriftzeichen, ASCII-Zeichen), eine Bitfolge generiert sowie eine weitere Bitfolge zur Fehlerkorrektur. Diese Bitfolge<sup>2</sup> wird nun in den noch freien Bereich eingesetzt, und zwar von unten rechts nach oben links, jeweils mit einem weißen Quadrat für eine „0“ und einem schwarzen Quadrat für eine „1“. Zur Vermeidung größerer schwarzer oder weißer Bereiche werden schließlich acht verschiedene Masken über den Text gelegt. Diejenige, welche das beste Ergebnis liefert, wird beibehalten und deren Kennnummer in den Code eingesetzt (ausführlich beschrieben auf <https://de.wikipedia.org/wiki/QR-Code>, weiterführend in der englischen Wikipedia: [https://en.wikipedia.org/wiki/QR\\_code](https://en.wikipedia.org/wiki/QR_code)).

Für weitere Informationen zum Thema QR-Codes können folgende Internetseiten besucht werden:

- <http://www.qrcode.com/en/> (Webseite der Firma *Denso Wave* zu QR-Codes, Englisch)
- <http://qrcode.wilkohartz.de>
- <http://wirtschaftslexikon.gabler.de/Definition/qrcode.html>
- <https://www.youtube.com/watch?v=qWgg1StZ2Kw>
- <https://www.youtube.com/watch?v=yiLjWbFQyF4>
- Freie QR-Code-Generatoren:
  - <http://www.qr-manager.com/qrcode-generator.html> (mit Fehlerkorrektur-Level)
  - <http://goqr.me/de/>
  - <http://www.qrcode-generator.de/>

In der Phase der Wortsammlung zum Begriff Informatik könnte eine *MindMap* ähnlich der folgenden Abbildung entstehen:

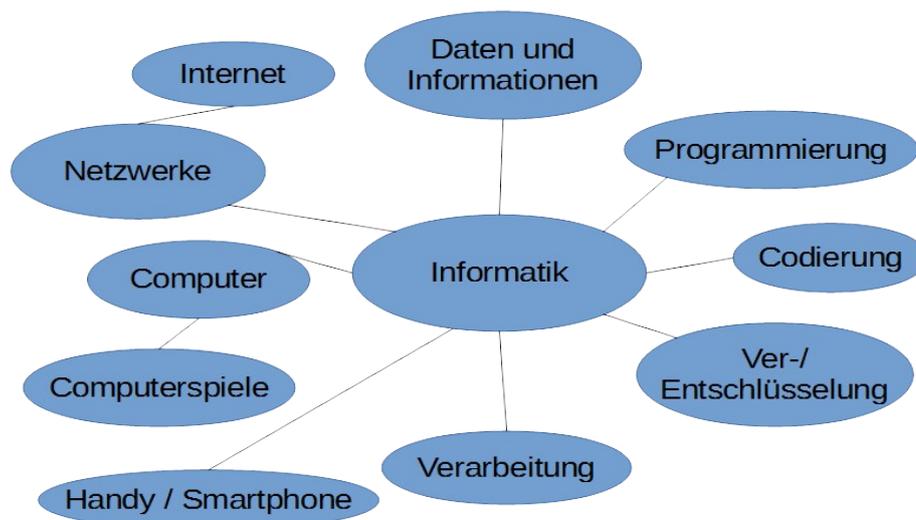
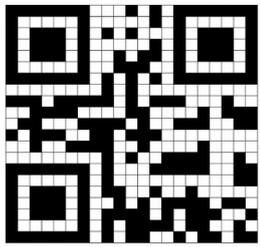
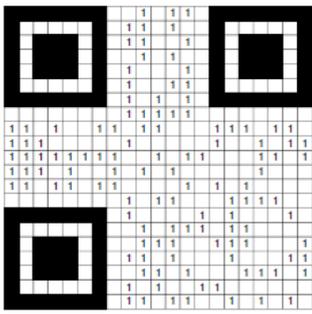
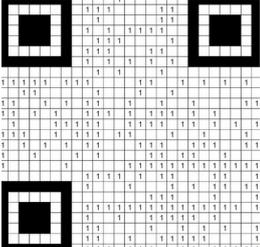


Abbildung 8: Beispiel für eine Wortsammlung zum Begriff Informatik

2 Bitfolge: Folge von Nullen und Einsen.



## Empfohlene Vorgehensweise für UE0: Einführung

Beschreibung des Unterrichtsgeschehens	Materialien
<ul style="list-style-type: none"> <li>Um eine Verbindung zur Lebenswirklichkeit der Kinder herzustellen, hängen Sie zur ersten Stunde einen QR-Code (z.B. „Informatik“ in <i>Kopiervorlage UE0.1</i>) an die Tafel.</li> <li>Sammeln Sie Ideen und Äußerungen der Kinder dazu. Eventuell bringen Sie Beispiele auf Verpackungen oder Werbung o. ä. mit.</li> <li>Lösen Sie den Code mit der Hilfe eines QR-Code-Scanners (z. B. mittels einer kostenfreien Smartphone-App) auf, oder lösen Sie das Wort „Informatik“ ohne Scanner auf. <i>Danach</i> nehmen Sie den ausgelesenen Begriff <i>Informatik</i> als Anfangsbegriff einer <i>Mind-Map</i>. Alternative Assoziationsketten sind ebenfalls möglich. Beispiel siehe .</li> <li>In der Wortsammlung sollte auch der Begriff <i>Computer</i> genannt werden.</li> <li>Fordern Sie die Schülerinnen und Schüler ausdrücklich auf, Fragen zum Thema zu stellen und halten Sie diese für den weiteren Verlauf der Unterrichtsreihe fest.</li> </ul>	 <p data-bbox="1107 647 1347 680">Kopiervorlage UE0.1</p>
<ul style="list-style-type: none"> <li>Tauschen Sie den an der Tafel hängenden Code gegen einen gleichen, mit unterlegtem Raster aus (<i>Kopiervorlage UE0.2</i>) und lassen Sie diesen von den Schülerinnen und Schülern beschreiben. Ziel ist es, dass die Kinder den Code als eine Anordnung von schwarzen und weißen Kästchen erkennen und somit deutlich wird, dass zwei Zustände ausreichen, um Daten zu codieren. Daher der Begriff <i>Digitale Welt</i>.</li> </ul>	 <p data-bbox="1107 1173 1347 1207">Kopiervorlage UE0.2</p>
<ul style="list-style-type: none"> <li>Teilen Sie an die Schülerinnen und Schüler vorbereitete QR-Codes (<i>Kopiervorlage UE0.3 a-p</i>) aus, bei denen alle mit 1 markierten Kästchen grob geschwärzt werden müssen (vorgeschlagene Sozialform Partnerarbeit). Den Schülerinnen und Schülern ist es an dieser Stelle möglich, mehrere QR-Codes zu bearbeiten, wobei entscheidend ist, dass jedes Team mind. einen Code vollständig einfärbt.</li> <li>Lesen Sie die erstellten Codes ein oder benutzen Sie die Lösungskarten (<i>Kopiervorlage UE0.5</i>) und zeigen Sie den Schülerinnen und Schülern die ausgelesenen Begriffe (z. B. <i>Schulhof</i>, <i>Kreide</i>, etc.).</li> </ul>	 <p data-bbox="1107 1561 1347 1594">Kopiervorlage UE0.3</p>
<ul style="list-style-type: none"> <li>Geben Sie als Hausaufgabe den vorbereiteten Code „Freue dich auf die digitale Welt“ zum Einfärben mit nach Hause (<i>Kopiervorlage UE0.4</i>).</li> </ul>	 <p data-bbox="1107 1874 1347 1908">Kopiervorlage UE0.4</p>



# UE1: Binärdarstellung



## Lernziel der Stunde

Die Kinder lernen die Binärdarstellung von Zahlen kennen. Sie erkennen, dass im vierstelligen Binärcode alle Dezimalzahlen von 0 bis 15 darstellbar sind.

Die Schülerinnen und Schüler

- (DW – K1)** beschreiben ihren persönlichen Umgang mit Informatiksystemen,
- (DW – K2)** erläutern das Prinzip der Dateneingabe und der Datenübertragung,
- (DW – K3)** begründen die Einschränkung auf die Zustände *Strom an* und *Strom aus*,
- (DW – K4)** erklären, wie Information in digitale Daten umgewandelt werden kann,
- (DW – K5)** führen die Umwandlung von Zahlen in Binärzahlen wie auch in umgekehrter Richtung selbst durch.

## Inhalt der Stunde

Die Kinder erarbeiten die Farbcodedarstellung der Zahlen von 0 bis 15. Im weiteren Verlauf übertragen sie die Farben **Rot** und **Blau** auf die Zustände *an* und *aus* (entspricht 1/0).

Die Schülerinnen und Schüler kennen aus ihrer Umgebung häufig nur die Dezimaldarstellung. Damit sie die Arbeitsweise eines Informatiksystems verstehen können, müssen sie jedoch die Binärdarstellung kennenlernen.

Mit der Einführung des Binärcodes für die Zahlen 0 bis 15 soll den Schülerinnen und Schülern das Prinzip von Information und Daten näher gebracht werden. Hierbei werden die Lernenden feststellen, dass es nicht möglich ist, ohne weiteres Wissen (Absprachen) zu entscheiden, wofür z. B. die Binärzahl 1001 steht. Den Kindern kann auch gezeigt werden, dass durch das Umwandeln in die Binärdarstellung keine Informationen verloren gehen.

Ein weiterer wichtiger Aspekt ist die Vorgehensweise beim Einstieg in die Problematik. Dies kann ganz vereinfacht mit dem Entwickeln eines Algorithmus verglichen werden, da die Schülerinnen und Schüler das Prinzip nach und nach auf mehrere Stellen ausweiten und damit das abstrakte Prinzip weiter verfeinern und verbessern. Dasselbe Vorgehen kommt auch in der Softwareentwicklung vor, denn auch hier wird etwas entwickelt, getestet und wenn nötig weiter modifiziert.

## Dauer

ca. 90 Minuten

## Materialien

- QR-Code Scanner
- Kopiervorlagen:
  - UE1.1 a-c
  - UE1.2 a-f
  - UE1.3
  - UE1.4 a-c
  - UE1.5
  - UE1.6
  - UE1.7 a-c
  - UE1.8
  - UE1.9
  - UE1.10 a-p
- **Rote** und **blaue** Stifte  
(eventuell Steckwürfel)



## Fachliche Analyse zur UE1: Binärdarstellung

Nach einer kurzen Wiederholung des vorherigen Themas *QR-Codes* werden die Binärzahlen behandelt. Binärzahlen sind Zahlenfolgen, die mittels der beiden Ziffern 0 und 1 dargestellt werden. Wie so man überhaupt zwei Zustände bzw. genau diese zwei Zahlen braucht, wird im Folgenden erläutert.

Der Computer bzw. auch alle weiteren Informatiksysteme arbeiten mit elektrischem Strom. Sie bestehen dabei aus vielen kleinen Schaltern, die *Transistoren* genannt werden und im wesentlichen den Strom auf- bzw. abdrehen. So bedeutet der Zustand 1, dass der *Strom an* ist und Zustand 0, dass der *Strom aus* ist. Mit Hilfe dieser beiden Zustände können Informationen codiert bzw. zuerst dargestellt werden.

Um dieses Konzept besser zu verstehen, werden zunächst „normale“ Zahlen codiert. Unter diesen „normalen“ Zahlen verstehen die meisten Menschen das Dezimalsystem, welches grundsätzlich aus nur 10 Ziffern besteht, nämlich der 0, 1, 2, ..., 9. Möchte man nun die nachfolgende Zahl beschreiben, so fängt man wieder bei 1 an und hängt eine 0 dran; das bedeutet, man braucht eine weitere Stelle im System. Dieses System wird als Dezimalsystem bzw. die Zahlen als Dezimalzahlen bezeichnet, weil ihre Basis 10 ist. Dieses Prinzip des [Stellenwertsystems](#) begegnet den Schülerinnen und Schülern bereits beim Zerlegen von Zahlen in Hunderter, Zehner und Einer.

Nun kann zu jeder weiteren Basis (als *Basis* werden die Zahlen bezeichnet, die die Wertigkeit der Stellen des Systems angeben) ein geeignetes System bzw. eine andere Darstellung gefunden werden. Für das nachfolgend betrachtete System wird die Basis 2 (das liegt daran, dass wir zwei Zustände benötigen, Erinnerung: *Strom aus* und *Strom an*) gewählt, und somit erhält man das Binärsystem.

Jede Dezimalzahl kann auch als Binärzahl dargestellt werden. Die Binärdarstellung einer Dezimalzahl besteht demnach nur aus den beiden Ziffern 0 und 1, also aus *zwei Zuständen* innerhalb eines *Stellenwertsystems*. Da man jede Dezimalzahl als eine Binärzahl darstellen kann und auch umgekehrt, gibt es Methoden, wie man diese Zahlen in das jeweils andere System umwandeln kann. Außerdem gehen keine Informationen verloren und auch alle Rechenarten, die man im Dezimalsystem anwenden kann, sind ebenfalls mit dem Binärsystem möglich.

Die im Stellenwertsystem zur Basis 10 angegebene Zahl  $26 = 2 \cdot 10^1 + 6 \cdot 10^0$

<b>Dezimalzahl</b>	2	6
<b>Wert</b>	$10^1$	$10^0$

soll im Folgenden in eine Binärzahl, also im Stellenwertsystem zur Basis 2, angegeben werden. Die Wertigkeit der einzelnen Stellen im Binärsystem sind die Zweierpotenzen, also:  $2^0 = 1$ ,  $2^1=2$ ,  $2^2=4$ ,  $2^3=8$ ,  $2^4=16$ ,...

Die Dezimalzahl 26 kann nun eindeutig als Summe dieser Zweierpotenzen dargestellt werden:

$$(26)_{10} = 16 + 8 + 2 = 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = (11010)_2$$

<b>Binärzahl</b>	1	1	0	1	0
<b>Wert</b>	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$

Das Umwandeln einer Dezimalzahl wurde in diesem Fall durch „strukturiertes Ausprobieren“ durchgeführt, es gibt jedoch auch eine Rechenanweisung, welche anhand eines Beispiels im Folgenden erläutert wird:



Man teile diese Zahl durch die gewünschte Basis, in diesem Fall die 2. Das Ergebnis ist 13 und der entstandene Rest ist 0. Nun teilt man die 13 durch 2 und erhält 6 mit Rest 1. Dies wird weitergeführt bis das Ergebnis 0 ist. Anschließend betrachtet man alle Reste (diese können in diesem Fall nur die Ziffern 0 oder 1 sein) von unten/hinten nach vorne und erhält somit seine Binärzahl, nämlich 11010, siehe auch folgende Tabelle.

Rechnung	Ergebnis	Rest
26 : 2	13	0
13 : 2	6	1
6 : 2	3	0
3 : 2	1	1
1 : 2	0	1

Das Umwandeln der Binärzahl in eine Dezimalzahl ist einfacher, wenn man sie von hinten nach vorne betrachtet. InformatikerInnen fangen immer bei Null an zu zählen, sodass die Ziffer ganz rechts an der nullten Stelle steht. Eins weiter nach links steht die Ziffer an der ersten Stelle, etc. Nun bildet man die Zweierpotenz mit der richtigen Stelle im Binärcode. Das bedeutet, dass an nullter Stelle eine  $2^0=1$  steht, an erster Stelle eine  $2^1=2$ , an zweiter Stelle eine  $2^2=4$ , usw. Nun sind die jeweiligen Ziffern an dieser Stelle noch wichtig. Diese Ziffer, die immer entweder 0 oder 1 ist, wird nun mit der jeweiligen Zweierpotenz multipliziert und anschließend insgesamt addiert. Folgendes Beispiel, dass  $11010 = 26$  ist, kann zur Verständigung beitragen.

Binärzahl	1	1	0	1	0
Wert	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$

Zugehörige Rechnung:

$$1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 16 + 8 + 0 + 2 + 0 = 26$$

Diese Betrachtungsweise funktioniert übrigens bei jeder Basis, auch im Dezimalsystem selbst:

Dezimalzahl	2	6
Wert	$10^1$	$10^0$

Zugehörige Rechnung:

$$2 \cdot 10^1 + 6 \cdot 10^0 = 20 + 6 = 26$$

Für die Kinder ist das Umrechnen einfacher, wenn man die Binärzahlen von links nach rechts betrachtet. Im Modul werden zur Vereinfachung für die Schülerinnen und Schüler weiter die Ergebnisse der Zweierpotenzen (16, 8, 4, 2, 1) angegeben. Außerdem werden die beiden Zustände 0 und 1 durch andere Arten dargestellt, beispielsweise durch die Farben der Karten (**rot** = 1 und **blau** = 0) oder durch die Lampe (an = 1 und aus = 0). So wird verdeutlicht, dass es sich bei der Binärdarstellung um ein allgemeines Konzept handelt, das üblicherweise über die *beiden Zustände* 0 und 1 dargestellt wird.

## Zusatzinformation

### Rechnen mit Binärzahlen

Analog zu den Rechenverfahren im Dezimalsystem kann man auch im Binärsystem schriftlich addieren und multiplizieren. Die Besonderheit liegt in diesem Fall nur darin, dass der Übertrag zur nächsten Stelle bereits bei der Addition von zwei Einsen entsteht.

Hierzu zwei Beispiele:



0110 1010 + 0011 1001 = ?

$$\begin{array}{r}
 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0 \\
 +\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 1 \\
 \hline
 \underline{1\ 1\ 1\ 1\ 1\ 1\ 1\ 1} \\
 1\ 0\ 1\ 0\ 0\ 0\ 1\ 1
 \end{array}$$

0110 · 0101 = ?

$$\begin{array}{r}
 0\ 1\ 1\ 0\ * \ 0\ 1\ 0\ 1 \\
 \hline
 \phantom{0\ 1\ 1\ 0\ * \ 0\ 1\ 0\ 1} 0\ 0\ 0\ 0 \\
 +\phantom{0\ 1\ 1\ 0\ * \ 0\ 1\ 0\ 1} 0\ 1\ 0\ 1 \\
 +\phantom{0\ 1\ 1\ 0\ * \ 0\ 1\ 0\ 1} 0\ 1\ 0\ 1 \\
 +\phantom{0\ 1\ 1\ 0\ * \ 0\ 1\ 0\ 1} 0\ 0\ 0\ 0 \\
 \hline
 0\ 0\ 1\ 1\ 1\ 1\ 0
 \end{array}$$

Wie im ersten Beispiel sichtbar wird, ist bei der Addition zweier Zahlen die Möglichkeit der Kombinationen an den einzelnen Stellen begrenzt:

Ziffer 1	Ziffer 2	Alter Übertrag	Ergebnis	Neuer Übertrag
0	0	0	0	0
1	0	0	1	0
0	1	0	1	0
1	1	0	0	1
0	0	1	1	0
1	0	1	0	1
0	1	1	0	1
1	1	1	1	1

Dies ist besonders relevant für die technische Umsetzung dieser Rechenart. Die Multiplikation kann auf die Addition zurückgeführt werden, hierbei muss für jede Stelle der ersten Zahl überprüft werden, ob ab dieser Stelle die zweite Zahl erneut addiert werden muss (1) oder nicht (0).

Die Subtraktion und Division können ebenfalls schriftlich durchgeführt werden. Im Computer wird jedoch die Subtraktion als Addition mit negativen Zahlen durchgeführt (siehe folgender Abschnitt) und die Division in der Folge als wiederholte Subtraktion.

### Negative Zahlen, Fließkommazahlen und Buchstaben

Negative ganze Zahlen werden im Computer nicht einfach als ihre positive Darstellung mit einem gesondert gespeicherten Vorzeichen betrachtet, sondern es wird ausgenutzt, dass der Computer nur eine begrenzte Speichergröße hat. Nehmen wir hierzu an, dass wir in unserem PC nur vier Stellen große Binärzahlen abspeichern können. Was passiert also wenn man 0001+1111 rechnet?

$$\begin{array}{r}
 0\ 0\ 0\ 1 \\
 +\ 1\ 1\ 1\ 1 \\
 \hline
 \underline{1\ 1\ 1\ 1} \\
 1\ 0\ 0\ 0\ 0
 \end{array}$$

Diese Zahl ist mit vier Stellen nicht mehr darstellbar; die vordere Stelle kann nicht gespeichert werden und somit ergibt diese Addition 0. Die Addition der Zahl 15 zeigt somit dasselbe Verhalten wie die Subtraktion von 1 bzw. die Addition von -1.

Dies wird genutzt, um das so genannte Zweierkomplement zu bilden. Der Zahlbereich wird hierbei in zwei Bereiche geteilt und alle Zahlen mit einer eins an erster Stelle werden nicht mehr als positive Binärzahl, sondern als ihr Zweierkomplement betrachtet. Für unser Beispiel sähe die neue Verteilung wie folgt aus:



Binärzahl	Dezimalzahl		Zweierkomplement	Dezimalzahl
0	0			
0001	1		1111	-1
0010	2		1110	-2
0011	3		1101	-3
0100	4		1100	-4
0101	5		1011	-5
0110	6		1010	-6
0111	7		1001	-7

Neben den negativen Zahlen erhalten auch die Fließkommazahlen eine besondere Darstellung. Nach anfänglichen Problemen haben sich die Chiphersteller hier auf einen einheitlichen Standard geeinigt, der in einem eigenen Bereich eines Chips verarbeitet wird, der so genannten *Floating-Point-Unit (FPU)*.

Für die Darstellung hat man sich auf ein Pendant zur wissenschaftlichen Darstellung von Dezimalzahlen geeinigt. In dieser wird beispielsweise die Zahl -1253,65 vereinheitlicht dargestellt als:  $-1,25365 \cdot 10^3$ , also so umgewandelt, dass eine Ziffer vor dem Komma steht und mithilfe einer Zehnerpotenz die Dimension der Zahl bestimmt. Die Zahl kann zerlegt werden in das *Vorzeichen*, die *Potenz* und die so genannte *Mantisse*, also die Zahlendarstellung, welche nacheinander gespeichert werden (im obigen Beispiel also 1,25365). Wie viele Stellen dazu für Mantisse und Potenz reserviert werden, hängt von dem gewählten Gesamtspeicherbedarf ab (weitere Informationen unter: [https://de.wikipedia.org/wiki/IEEE\\_754](https://de.wikipedia.org/wiki/IEEE_754)).

Für Buchstaben oder auch die Codierung von Bildern wurde sich, ähnlich wie bei den Fließkommazahlen, darauf geeinigt, welche Bedeutung einzelne Stellen oder Blöcke innerhalb von Zahlenkolonnen haben, z. B. in Form des *ASCII-Codes* (siehe hierzu auch den Fachtext zum Thema Datenübertragung).

### Bits & Bytes

Wie bereits erwähnt, speichert der Computer alle Daten in Form von Zahlen ab, die zudem in Blöcken bestimmter Länge angeordnet beziehungsweise verrechnet werden. Die kleinste Einheit, also eine einzelne Speicherzelle, die nur eine einzelne 1 oder 0 speichern kann, nennt man hierbei *Bit*. Diese Einheit begegnet einem vor allem bei Übertragungsraten, wie z. B. beim WLAN (mögliche Geschwindigkeit: 150 Mbit/s). Historisch bedingt werden in den Prozessoren Vielfache von 8 Bits gleichzeitig verarbeitet (momentan bis zu 64 Bit). Diese 8 Bit nennt man zusammengefasst dann auch ein *Byte*.

Auf Festplatten findet sich dabei z. B. die Angabe: 500 GB, also 500 Giga-Byte. Hierbei ist wie gewohnt:

- 1 KB = 1 Kilobyte = 1000 Byte
- 1 MB = 1 Megabyte = 1000 Kilobyte = 1000 · 1000 Byte = 1 000 000 Byte
- 1 GB = 1 Gigabyte = 1000 Megabyte = 1000 · 1000 · 1000 Byte = 1 000 000 000 Byte
- 1 TB = 1 Terabyte = 1000 Gigabyte = 1000 · 1000 · 1000 · 1000 Byte = 1 000 000 000 000 Byte

Da diese Einheiten in der Informatik nicht immer praktikabel sind (da alles in Potenzen von 2 vorkommt), gibt es in einigen Bereichen, z. B. im Bereich von Arbeitsspeichern, auch alternative Präfixe, welche nicht in 1000er-Schritten, sondern in 1024er-Schritten ( $1024 = 2^{10}$ ) vorgehen. Diese werden mit einem kleinen „i“ gekennzeichnet:



- 1 KiB = 1 Kibibyte = 1024 Byte
- 1 MiB = 1 Mebibyte = 1024 Kibibyte = 1024 · 1024 Byte = 1048576 Byte
- 1 GiB = 1 Gibibyte = 1024 Mebibyte = 1024 · 1024 · 1024 Byte = 1073741824 Byte

### Weitere Zahlensysteme

Neben dem Binär- und dem Dezimalsystem gibt es noch weitere Systeme, da zu jeder natürlichen Zahl (ohne Null) ein eigenes Stellenwertsystem erstellt werden kann. Selbst zur Basis 1 gibt es das *Unärsystem* (meist als Strichliste realisiert).

Einige heute nicht mehr gebräuchliche Zahlensysteme haben ihre Spuren in unserem Alltag hinterlassen. Zum Beispiel das *Duodezimalsystem* (zur Basis 12) durch den Begriff „ein Dutzend“, oder die Stundeneinteilung unserer Uhr. Die Minuteneinteilung wiederum erfolgt aus dem *Sexagesimalsystem* (zur Basis 60) der Babylonier.

Im Bereich der Informatik wurde früher auch noch das *Ternärsystem* (zur Basis 3) genutzt, indem man neben den Zuständen 1 und 0 („Strom fließt“ = positive Spannung / „Strom fließt nicht“ = keine Spannung) noch den Zustand -1 einführte („Strom fließt in die andere Richtung“ = negative Spannung).

Heutzutage werden jedoch meist nur noch Zahlensysteme mit Zweierpotenzen als Basis verwendet. Neben dem bereits kennengelernten Binärsystem sind das insbesondere das *Oktalsystem* (zur Basis 8), welches z. B. genutzt wird, um Dateiberechtigungen zu verwalten, und das *Hexadezimalsystem* (zur Basis 16) um Farben, insbesondere bei der Erstellung von Websites zu codieren.

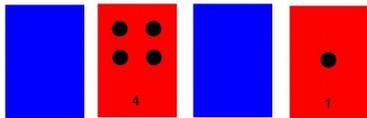
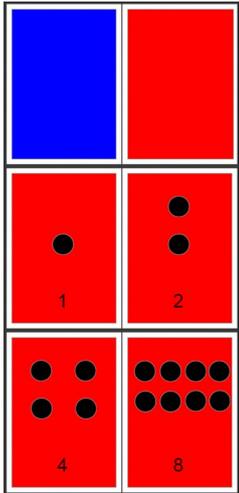
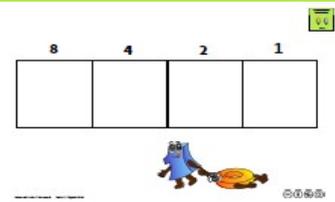
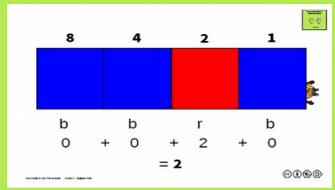
Der besondere Vorteil dieser Zahlensysteme ist, dass diese einfach ineinander überführt werden können. So können zum Umrechnen aus dem Binärsystem ins Oktalsystem immer drei Stellen der Binärzahl direkt in eine Stelle der Oktalzahl umgewandelt werden. Genauso ist es beim Hexadezimalsystem, wobei hier immer vier Stellen in eine Stelle umgewandelt werden können.

Da das Hexadezimalsystem 16 verschiedene Ziffern verwendet, werden hierbei neben den Ziffern 0 bis 9 noch die Buchstaben A bis F für die Wertigkeiten von 10 bis 15 verwendet. Somit kann beispielsweise die Binärzahl 0111 1011 (Dezimal: 133) in die Hexadezimalzahl 7B umgewandelt werden.

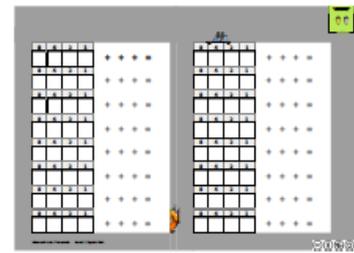
Weitere Informationen finden sich auf folgenden Internetseiten:

- <http://www.ulthryvasse.de/binaerzahlen.html>
- <https://www.youtube.com/watch?v=6Wsl95N0QKU>

## Empfohlene Vorgehensweise für UE1: Binärdarstellung

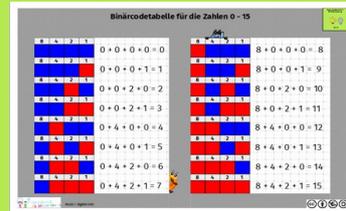
Beschreibung des Unterrichtsgeschehens	Materialien
<p>Falls Sie die Stunde UE0 durchgeführt haben, ist dieses ein möglicher <u>Einstieg</u>:</p> <ul style="list-style-type: none"> <li>Hängen Sie drei QR-Codes an die Tafel, die die Zahlen 5, 9 und 14 darstellen (<i>Kopiervorlage UE1.1 a-c</i>). Lesen Sie die QR-Codes aus (oder nennen Sie die Ergebnisse) und zeigen Sie den Kindern die ausgelesenen Zahlen.</li> <li>Erarbeiten Sie im Unterrichtsgespräch, dass diese Form der Codierung sehr umständlich ist, da für eine einfache Dezimalzahl viele Quadrate gezeichnet werden müssen.</li> </ul>	 <p>Kopiervorlage UE1.1 a-c</p>
<p><u>Alternative Einstiegsstelle:</u></p> <ul style="list-style-type: none"> <li>Bitten Sie vier Kinder nach vorne und statten Sie diese mit Zahlen-/Punktearten (8, 4, 2, 1) aus (<i>Kopiervorlage UE1.2</i>). Die Kinder zeigen zunächst ihren Mitschülerinnen und Mitschülern die <b>blauen</b> Seiten ihrer Karten.</li> <li>Geben Sie den Kindern die Aufgabe, der Klasse zunächst die Zahl 4 (<b>blau</b>, <b>rot 4</b>, <b>blau</b>, <b>blau</b>), daraufhin die Zahlen 8 (<b>rot 8</b>, <b>blau</b>, <b>blau</b>, <b>blau</b>), 1 (<b>blau</b>, <b>blau</b>, <b>blau</b>, <b>rot 1</b>) oder ähnliches zu zeigen.</li> <li>In der nächsten Aufgabe fordern Sie die Kinder auf, mit ihren Karten die Zahl 10 (<b>rot 8</b>, <b>blau</b>, <b>rot 2</b>, <b>blau</b>) zu zeigen. Damit wird den Schülerinnen und Schülern klar, dass mit Hilfe der vier Karten auch größere Zahlen als 8 dargestellt werden können.</li> </ul> <p>Beispiel 5:</p> 	 <p>Kopiervorlage UE1.2</p>
<ul style="list-style-type: none"> <li>Wenn das Prinzip der Darstellung verstanden worden ist, bitten Sie vier andere Kinder nach vorne, die Sie mit vier neuen Karten (nur <b>blaue</b> und <b>rote</b> Seite – keine Zahlendarstellung) ausstatten (<i>ebenfalls Kopiervorlage UE1.2</i>). Fordern Sie die Kinder auf, verschiedene Zahlen darzustellen (z. B. 13, 11, etc.). Die Kinder finden heraus, dass mit vier Stellen und zwei Zuständen 16 verschiedene Zahlen darstellbar sind.</li> <li>Im Folgenden sollen die Schülerinnen und Schüler alle mit vier Stellen und zwei Zuständen darzustellenden Zahlen entwickeln. Um den Kindern die Regeln der Codierung zu erklären, hängen Sie die leere Vorlage (<i>Kopiervorlage UE1.3</i>) an die Tafel.</li> <li>Erläutern Sie den Kindern folgende Regeln:             <ol style="list-style-type: none"> <li>Alle Stellen können mit <b>blau</b> oder <b>rot</b> besetzt werden.</li> <li>Alle Stellen müssen besetzt werden.</li> <li>Eine mit <b>rot</b> besetzte Stelle wird gezählt (1).</li> <li>Eine mit <b>blau</b> besetzte Stelle wird nicht gezählt (0).</li> </ol> </li> <li>Erarbeiten Sie zur Verdeutlichung mit den Schülerinnen und Schülern an der Tafel ein paar Beispiele (<i>Kopiervorlage UE1.4</i>).</li> </ul>	 <p>Kopiervorlage UE1.3</p>  <p>Kopiervorlage UE1.4 a-c</p>

- Teilen Sie an die Schülerinnen und Schüler eine leere *Binärcodetabelle* (*Kopiervorlage UE1.5*) aus und fordern Sie sie auf, mit **blauem** und **rotem** Stift auf der Vorlage unter möglichst systematischem Vorgehen alle im *Binärcode* mit vier Stellen darzustellenden Zahlen zu suchen, einzufärben, zu berechnen und zu notieren. Alternativ können zur Darstellung der Zahlen auch **blaue** und **rote** Steckwürfel verwendet werden (vorgeschlagene Sozialform Partnerarbeit).  
Bei schnellen Gruppen können Sie diese Zusatzaufgabe stellen: *Überlegt euch eine Darstellung der Zahl 20.*



Kopiervorlage UE1.5

- Um die Schülerergebnisse zu sichern, lassen Sie nach der Arbeit die Schülerinnen und Schüler einzelne Zahlen und ihre Codierung nennen, tragen Sie diese mit farbigen Stiften an die richtige Stelle der Tabelle ein und notieren Sie dahinter die dazugehörige Rechnung sowie die Zahl (mögliche Zahlen 0 bis 15).
- Wenn Sie mit den Kindern alle Zahlen entwickelt und in die *Codetabelle* eingetragen haben, ersetzen Sie diese durch die farbige *Binärcodetabelle* (*Kopiervorlage UE1.6*).

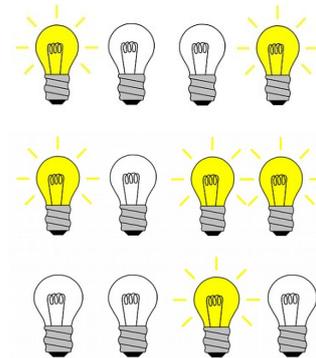


Kopiervorlage UE1.6

**Bemerkung:**

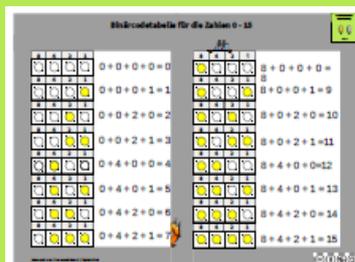
Wenn Sie den Erarbeitungsprozess verkürzen wollen, können Sie auch das **Blau/Rot**-System weglassen und direkt mit dem An/Aus-System starten. Hierbei lassen Sie die Kinder *Kopiervorlage UE1.5* nur mit einem **gelbem** Stift bearbeiten. Um einen ähnlich aktiven Einstieg zu ermöglichen, könnte die Glühlampenkarten von UE2 (*Kopiervorlage UE2.2*) benutzt werden.

- Im Folgenden sollen die Kinder die beiden Zustände **blau** und **rot** auf die Zustände *an* und *aus* transferieren.
- Dazu zeigen Sie den Schülerinnen und Schülern drei Bildbeispiele (*Kopiervorlage UE1.7*) mit leuchtenden und nicht leuchtenden Glühlampen. Es ist zu erwarten, dass die Schülerinnen und Schüler durch den Impuls die Analogie zu den Farben selbstständig herstellen. Gegebenenfalls müssen von Ihnen im Unterrichtsgespräch noch Hilfestellungen gegeben werden (**rot** = Lampe leuchtet, **blau** = Lampe leuchtet nicht).
- Thematisieren Sie im Gespräch mit den Schülerinnen und Schülern deutlich, warum gerade die Zustände „Strom an“ und „Strom aus“ die Arbeitsweise eines Computers (Tablets, Smartphones etc.) verdeutlichen.



Kopiervorlage UE1.7 a-c

- Ist den Schülerinnen und Schülern diese Analogie deutlich geworden, ersetzen Sie als Lehrkraft die farbige *Binärcodetabelle* durch die veränderte *Binärcodetabelle* (*Kopiervorlage UE1.8*).
- Damit die Lernenden die Erkenntnisse der Unterrichtsstunde anwenden können, führen Sie mit ihnen ein vorbereitetes *Bingospiele* (*Kopiervorlage UE1.9*) mit 3x3 Feldern durch. Dabei werden die *Bingozahlen* (0 bis 15) mit Hilfe von laminierten gemischten Bildkarten (*Kopiervorlage UE1.10*) zufällig gezogen.



Kopiervorlage UE1.8



# UE2: Datenübertragung



## Lernziel der Stunde

Die Kinder lernen, dass sie mit einem fünfstelligen Binärcode Buchstaben codieren und somit Textnachrichten übertragen können. Sie entdecken bei der Anwendung die Äquivalenz von 1/0 zu *an/aus*.

Die Schülerinnen und Schüler

**(DW – K2)** erläutern das Prinzip der Dateneingabe und der Datenübertragung,

**(DW – K3)** begründen die Einschränkung auf die Zustände *Strom an* und *Strom aus*,

**(DW – K4)** erklären, wie Information in digitale Daten umgewandelt werden kann,

**(DW – K5)** führen die Umwandlung von Zahlen und Buchstaben in Binärzahlen wie auch in umgekehrter Richtung selbst durch.

## Inhalt der Stunde

Anhand von Bildkarten mit dargestellten Leuchtmitteln können die Kinder Buchstaben identifizieren. Diese Erkenntnis verwenden sie beim Versenden und Empfangen von Botschaften.

Über die Codierung der Buchstaben als Binärstring wird den Schülerinnen und Schülern zum einen das Prinzip der Daten und Information und zum anderen das Grundprinzip des Codes für die Binärdarstellung von Buchstaben und Satzzeichen beigebracht (vgl. ASCII-Code). Damit lernen die Schülerinnen und Schüler auch die Grundlage des Computers kennen, da dieser nur den Binärcode verarbeiten kann. Die Kinder müssen nun entscheiden, wofür die Zahl 1001 stehen kann.

Sie lernen auch ganz rudimentär das EVA-Prinzip (Eingabe-Verarbeitung-Ausgabe) kennen. Die *Eingabe* entspricht der nicht codierten Nachricht. Die *Verarbeitung* übernehmen die Schülerinnen und Schüler mittels eines Quasi-Algorithmus, der das Umwandeln von einer Nachricht in Binärstrings tätigt. Die *Ausgabe* ist dann der Binärstring, in welchem die komplette Nachricht steht. Hierbei lernen die Kinder, dass sie (bzw. Computer) sich präzise und eindeutig ausdrücken müssen.

## Dauer

ca. 90 Minuten

## Materialien

- Kopiervorlagen:
  - UE2.1
  - UE2.2
  - UE2.3 (32 Karten)
  - UE2.4
  - UE2.5
  - UE2.6
  - UE2.7
- Gelbe Stifte, Klebstoff, Scheren

## Fachliche Analyse zur UE2: Datenübertragung

### Digitale Datenübertragung

Für Computer wird ein Stellenwertsystem benutzt, in dem es nur die Ziffern 0 und 1 gibt, das sogenannte *Binärsystem* (siehe Fachliche Analyse zur Unterrichtseinheit 1: Binärdarstellung). Aus diesem Grund verläuft auch die Datenübertragung nur mittels dieser beiden Ziffern. Dies bedeutet, dass alle zu übertragenden Informationen zunächst binär codiert werden müssen. Bei der Übertragung eines Textes sind demnach alle Satzzeichen, alle Ziffern und alle Buchstaben in eine Binärzahl umzuwandeln, bevor der Computer diese Daten verarbeiten und verschicken kann. Eine mögliche Darstellung der Zeichen als Binärzahlen ist der ASCII-Code. Durch den ASCII-Code wird jedem Zeichen (Buchstabe, Ziffer und Sonderzeichen) eine Bitfolge aus 7 Bits zugeordnet. Da jedes Bit den Wert 0 oder 1 annehmen kann, können  $2^7 = 128$  verschiedene Zeichen eindeutig mit Hilfe des Binärsystems dargestellt werden. ASCII-Tabellen<sup>3</sup> geben darüber Aufschluss, welche Binärcodierung für das jeweilige Zeichen vorgesehen ist. Da es jedoch weit mehr als 128 verschiedene Zeichen gibt, wurde der ASCII-Code erweitert, indem das siebenstellige Bitmuster um ein Bit verlängert wurde.

Schriftzeichen	Dezimal	Hexadezimal	Binär
A	65	41	(0)1000001
B	66	42	(0)1000010
C	67	43	(0)1000011
...	...	...	...

Mit Hilfe einer Tabelle (*Kopiervorlage UE2.1*) können die Schülerinnen und Schüler Buchstaben und Satzzeichen in das Binärsystem übertragen, bzw. auch den umgekehrten Weg übernehmen. Diese Tabelle entspricht grundsätzlich einer ASCII-Tabelle, in der zu jedem Buchstaben bzw. Satzzeichen ein Binärcode angegeben wird. Das bedeutet, dass die Schülerinnen und Schüler Buchstaben oder Satzzeichen erhalten (Eingabe), diese mit Hilfe der ASCII-Tabelle in Binärzahlen umwandeln (Verarbeitung), um schließlich das ursprüngliche Zeichen in binärer Form darzustellen (Ausgabe). Dieses Vorgehen wird in der Informatik in vielen Bereichen angewendet und nennt sich EVA-Prinzip (Abbildung unten).



Ein Beispiel für das EVA-Prinzip ist die Darstellung des Zeichens auf dem Monitor, dessen Taste gedrückt wurde:



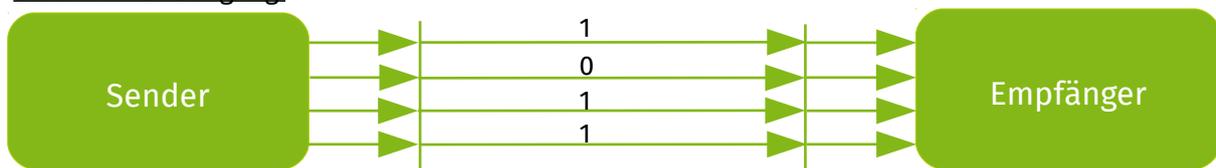
3 Es gibt viele Internetseiten, die vollständige ASCII-Tabellen bereitstellen, z.B. [www.asciitable.com](http://www.asciitable.com).

Nachdem die Daten codiert wurden, können sie physikalisch auf verschiedene Weisen vom Sender zum Empfänger übertragen werden. Die Übertragung erfolgt durch elektrische oder optische Signale oder elektromagnetische Wellen. Werden elektrische Signale zur Datenübertragung verwendet, kann anhand der elektrischen Spannung bzw. des elektrischen Stroms festgestellt werden, ob eine 0 oder eine 1 übertragen wird. Bei optischen Signalen dient Licht der Übertragung von Nullen und Einsen. Hinsichtlich der elektrischen Signale werden die positive und die negative Logik unterschieden in Abhängigkeit davon, mit welcher Spannung 0 und 1 jeweils übertragen werden. Gemäß der positiven Logik wird dabei die 1 durch einen hohen Spannungswert und die 0 durch einen niedrigen Spannungswert repräsentiert; bei der negativen Logik ist die Zuordnung umgekehrt. Je nach Übertragungsverfahren werden die einzelnen Bits eines Zeichens nacheinander (serielle Übertragung) oder gleichzeitig (parallele Übertragung) übertragen. Die simultane Übertragung von mehreren Zeichen wird dabei durch die Verwendung entsprechend vieler Kanäle erreicht.

Serielle Übertragung:



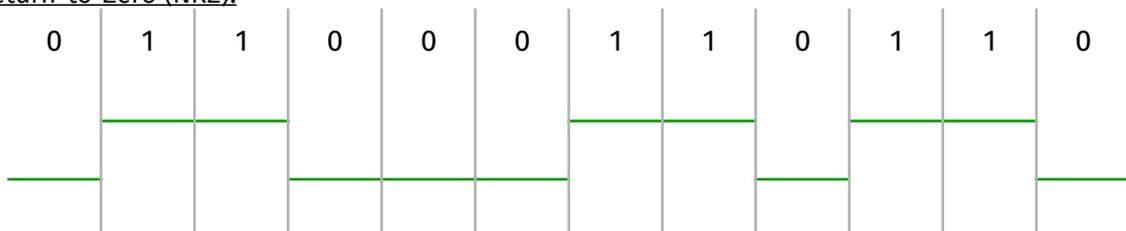
Parallele Übertragung:



**Zusatzinformation**

Auf zeitlicher Ebene wird ein Taktsignal genutzt, damit der Empfänger aufeinander folgende Bits gleichen Wertes differenzieren kann. Ein Beispiel für eine serielle Übertragung von Bits unter Verwendung von positiver Logik und einem Taktsignal ist die *NRZ-Codierung*.

Non-Return-to-Zero (NRZ):



Da die Taktsignale im Abstand von nur wenigen Nanosekunden aufeinander folgen, ist es bei einer Reihung desselben Bits schwierig, die Länge der Bitfolge zu bestimmen, da sich der Spannungswert bei der NRZ-Kodierung nur ändert, wenn eine 0 auf eine 1 bzw. eine 1 auf eine 0 folgt. Deshalb werden in der Praxis Codierungen mit häufigem Spannungswechsel verwendet. Zusätzlich werden Strategien zur Fehlererkennung und -behebung angewandt, um eine fehlerfreie Datenübertragung zu gewährleisten.

Zur Übertragung optischer Signale werden Glasfaserkabel verwendet. Diese sind weniger störänfällig (d. h. Daten können auch über sehr große Distanzen hinweg fehlerfrei übertragen werden) und ermöglichen eine besonders schnelle Datenübertragung.

In Abhängigkeit von der Sender- bzw. Empfängeranzahl und der Durchlässigkeit der Verbindung zwischen ihnen, werden zudem folgende Übertragungsarten unterschieden:

Bei einer *Punkt-zu-Punkt-Verbindung* tauschen genau ein Sender und ein Empfänger Daten aus. Kommuniziert hingegen ein Sender mit mehreren Empfängern oder besteht die Möglichkeit, dass verschiedene Sender demselben Empfänger Daten schicken, spricht man von einer *Punkt-zu-Mehrpunkt-Verbindung*.

Über eine *Simplex-Verbindung* können Daten nur unidirektional übertragen werden.

Als *Halbduplex* bezeichnet man die Verbindungsart, bei der Daten zwar prinzipiell in beide Richtungen übertragen werden können, dies jedoch nicht simultan, sondern nur abwechselnd möglich ist. Eine *Duplex-Verbindung* ermöglicht es, dass Daten gleichzeitig in beide Richtungen übertragen werden können.

Die untenstehenden Abbildungen dienen der Veranschaulichung der verschiedenen Übertragungsarten.

Beispiel	Übertragungsart
 <p>Abbildung 12: Dosentelefon</p>	<p><u>Punkt-zu-Punkt</u> Beim Telefonieren tauschen <b>ein</b> Sender und <b>ein</b> Empfänger Nachrichten aus.</p>
 <p>Abbildung 13: Blaulicht</p> <p><b>SMART HOME</b></p>  <p>Abbildung 14: Smarthome-Icons</p>	<p><u>Punkt-zu-Mehrpunkt</u> Schaltet ein Feuerwehrauto im Straßenverkehr Blaulicht und Martinshorn ein, geht das Signal von <b>einem einzelnen Sender</b> (dem Feuerwehrauto) aus. Empfangen wird das Signal jedoch von <b>vielen verschiedenen Empfängern</b> (z. B. von allen Verkehrsteilnehmern, die sich in der Nähe des Feuerwehrautos befinden).</p> <p>Umgekehrt senden beispielsweise mehrere verschiedene Sensoren Daten wie die Raumtemperatur an das Steuerungsmodul eines <i>Smart-Home-Systems</i>.</p>

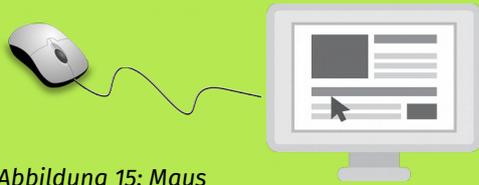


Abbildung 15: Maus



Abbildung 16:  
Bildschirm

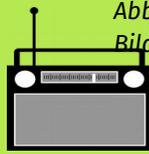


Abbildung 17: Radio



Abbildung 18:  
Fernseher

### Simplex

Die Maus sendet ihre Bewegungsdaten an den Computer, wodurch die Position des Cursors auf dem Bildschirm bestimmt werden kann. Umgekehrt sendet der Computer jedoch keine Daten an die Maus.

Wenn man fernsieht oder Radio hört, **empfängt** man Daten (Bild und Ton), **kann selbst jedoch keine Daten senden**.



Abbildung 19:  
Gegensprech-  
anlage



Abbildung 20:  
Taschenlampe

### Halbduplex

Tauschen zwei Kommunikationspartner per optischem Signal Morsezeichen aus, so agieren sie abwechselnd als Sender und Empfänger. Es ist jedoch **nicht möglich**, einen Code aus kurzen und langen Zeichen **zu empfangen und gleichzeitig** selbst einen solchen zu **senden**.

Gegensprechanlagen wie sie z. B. am Eingang von Mehrfamilienhäusern zu finden sind, nutzen ebenfalls eine Halbduplex-Verbindung.

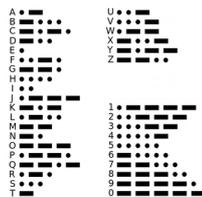


Abbildung 21:  
Morsecode



Abbildung 22: Telefonieren

### Duplex

Ein Telefonat zwischen einem Sender und einem Empfänger ist nicht nur ein Beispiel für eine Punkt-zu-Punkt-Verbindung, sondern auch für eine Duplex-Verbindung, da **beide** Kommunikationspartner **gleichzeitig** sprechen und den anderen hören können, z. B. wenn einer dem anderen ins Wort fällt.

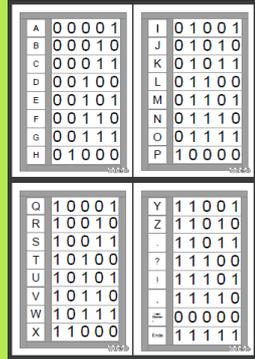
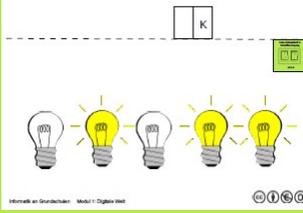
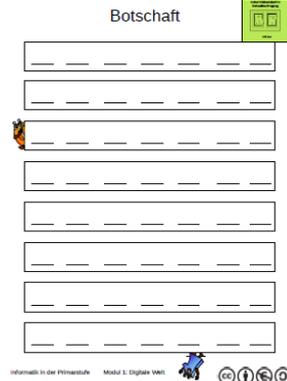
Weitere Informationen zum EVA-Prinzip: <https://de.wikipedia.org/wiki/EVA-Prinzip>



### Literatur:

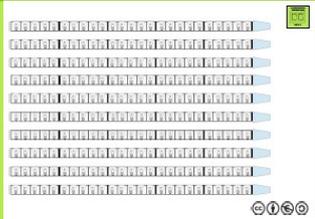
- <http://www.elektronik-kompodium.de/sites/kom/0212091.htm>
- <http://www.informatik.uni-hamburg.de/TKRN/world/lernmodule/Lmint/Popup/datenuebertragen.htm>
- <http://www.itwissen.info/definition/lexikon/Uebertragungsart-transmission-mode.html>
- <http://www.itwissen.info/definition/lexikon/Datenuebertragung-DT-data-transmission-Due.html>
- <http://www.itwissen.info/definition/lexikon/Manchester-Codierung-Manchester-encoding.html>

## Empfohlene Vorgehensweise zur UE2: Datenübertragung

Beschreibung des Unterrichtsgeschehens	Materialien
<ul style="list-style-type: none"> <li>• Zum Einstieg in die Unterrichtsstunde hängen Sie unkommentiert die <i>Codetabelle (Kopiervorlage UE2.1)</i> mit Buchstaben und Satzzeichen an die Tafel.</li> <li>• Bitten Sie fünf Kinder nach vorne und übergeben Sie jedem eine Bildkarte mit „an“ bzw. „aus“ auf den beiden Seiten (<i>Kopiervorlage UE2.2</i>).</li> <li>• Stellen Sie den Kindern die Aufgabe, ihren Mitschülerinnen und Mitschülern mit Hilfe der Bildkarten verschiedene Buchstaben anzuzeigen.</li> <li>• Zur Vertiefung können Sie die Übung mit anderen Schülerinnen und Schülern wiederholen.</li> <li>• Zur weiteren Verdeutlichung zeigen Sie den Lernenden einige vorgefertigte Buchstaben- und Satzzeichenkarten (<i>Kopiervorlage UE2.3</i>). Die Schülerinnen und Schüler vergleichen die Bildtafeln mit den 0/1-Folgen der Codetabelle und finden erwartungsgemäß die dargestellten Buchstaben.</li> <li>• Notieren Sie die gefundenen Buchstaben an der Tafel. Weisen Sie besonders auf die Sonderzeichen (Satzzeichen, Leerzeichen, Satzende) der Codetabelle hin.</li> </ul>	 <p style="text-align: center;">Kopiervorlage UE2.1</p>  <p style="text-align: center;">Kopiervorlage UE2.3 (Beispiel „K“, insgesamt 32 Karten)</p>
<ul style="list-style-type: none"> <li>• Wenn das Prinzip der Darstellung klar ist, teilen Sie den Schülerinnen und Schülern ein Blatt zur Notation der Botschaft (<i>Kopiervorlage UE2.4</i>) und eine Minicodetabelle (<i>Kopiervorlage UE2.5</i>) aus.</li> </ul> <p><u>Empfohlene Sozialform:</u> Partnerarbeit oder Kleingruppe</p> <ul style="list-style-type: none"> <li>• Halten Sie einzelne Bildkarten hoch. Die zugehörigen Buchstaben werden von den Schülerinnen und Schülern aus der Codetabelle an der Tafel oder der Minicodetabelle entnommen und auf dem Blatt notiert. So entsteht eine kleine Botschaft an die Kinder. Wichtig: Denken Sie sich vorher eine lustige Botschaft an Ihre Klasse aus.</li> </ul>	 <p style="text-align: center;">Kopiervorlage UE2.4</p>
<ul style="list-style-type: none"> <li>• Im weiteren Verlauf der Unterrichtsstunde sollen die Schülerinnen und Schüler mit einem <i>Paperlaptop (Kopiervorlage UE2.6)</i> eine Botschaft schreiben und die zugehörige Codierung der Buchstaben und Satzzeichen auf einem Leuchtmittelstreifen (<i>Kopiervorlage UE2.7</i>) mit <b>gelbem</b> Stift einzeichnen.</li> <li>• Dazu erhalten die Arbeitsgruppen von Ihnen einen <i>Paperlaptop</i> und ein Blatt mit <i>Leuchtmittelstreifen</i>, die auseinander geschnitten und aneinander geklebt werden sollen. Die Schülerinnen und Schüler notieren auf ihrem <i>Papiercomputer</i> eine Nachricht, lesen den zugehörigen 1/0-Code auf der Tastatur ab und übertragen ihn mit <b>gelbem</b> Stift auf die Leuchtmittelstreifen, indem sie die dargestellten Glühbirnen <b>weiß</b> lassen oder <b>gelb</b> einfärben.</li> <li>• Die entwickelten und auf dem Leuchtmittelstreifen durch an/aus dargestellten Botschaften werden von den Schülerinnen und</li> </ul>	 <p style="text-align: center;">Kopiervorlage UE2.6</p>



Schülern an andere Gruppen versandt. Die Empfänger haben die Aufgabe, die weitergegebene Nachricht mit Hilfe ihrer Codetabelle oder Tastatur des *Paperlaptops* zu decodieren.



Kopiervorlage UE2.7

- Thematisieren Sie zum Abschluss der Unterrichtsstunde die bei Codierung und Decodierung aufgetretenen Probleme. Beziehen Sie auch die praktische Anwendung mittels Computern (u. U. auch in Form von Robotern o. ä. in die Diskussion ein).



# UE3: Fehlererkennung



## Lernziel der Stunde

Die Kinder lernen die Notwendigkeit einer einfachen Fehlererkennung bei der Datenübertragung kennen. Sie entdecken die Funktion einer mitgesendeten Kontrollzahl (Prüfbit).

Die Schülerinnen und Schüler

**(DW – K5)** führen die Umwandlung von Zahlen und Buchstaben in Binärzahlen wie auch in umgekehrter Richtung selbst durch,

**(DW – K6)** erläutern die Notwendigkeit von Fehlererkennungsmechanismen bei der Datenübertragung und

**(DW – K7)** beschreiben ein vorgegebenes Verfahren zur Lokalisierung einer fehlerhaften Stelle und wenden dies an.

## Inhalt der Stunde

Anhand einer fehlerhaften Botschaft machen die Schülerinnen und Schüler die Erfahrung, dass einzelne falsch übermittelte Ziffern ganze Nachrichten verfälschen können. Die Kinder erarbeiten die Funktion einer Kontrollziffer und wenden diese bei Codierung und Decodierung an. Im weiteren Verlauf nutzen sie ihr Wissen und *programmieren* einen Weg durch das Labyrinth.

Ganz besonders in dieser Doppelstunde wird ein Bezug zur *realen* Welt hergestellt. Anhand des Fehlers in dem Code und mit Hilfe des Beispiels des Roboters können sehr gut die Wechselwirkungen zwischen Informatik und der Gesellschaft dargestellt werden. Durch die Einführung des Prüfbits wird den Schülerinnen und Schülern ein gängiges System zur Fehlererkennung eröffnet.

## Dauer

ca. 90 Minuten

## Materialien

- Kopiervorlagen:
  - UE3.1
  - UE3.2
  - UE3.3
  - UE3.4
  - UE3.5
  - UE3.6
- Spielfiguren mit Gesichtern (und eindeutiger Bewegungsrichtung)

## Fachliche Analyse zur UE3: Fehlererkennung

Zu übermittelnde Daten werden mit Hilfe einer sogenannten *Kontrollziffer* gesichert. Fehler in den Daten können so mit Hilfe der Kontrollziffer direkt entdeckt werden. Diese Sicherung dient dazu, fehlerhafte Daten zu erkennen und auch Fehler bei der Übertragung zu bemerken. Auch können Eingabefehler somit vermieden werden.

Beim Übertragen der Daten kann es zu *Störungen* kommen. Dadurch können Ziffern in der Sequenz verändert werden (von 1 zu 0 oder umgekehrt) oder aber auch komplett verloren gehen. Werden solche Verfälschungen nicht erkannt, kann es passieren, dass diese fehlerhaften Daten weiterverarbeitet werden. Je nachdem können auf diese Weise Rechnungen oder Nachrichtenübertragungen nicht korrekt ausgewertet werden und die komplette Übertragung muss wiederholt werden, falls bekannt ist, dass der Wert (die Nachricht) nicht stimmen kann.

*Fehlerursachen* bei der Übertragung können Rauschen, Kurzzeitstörungen, Signalverformungen oder auch Nebensprechen sein. *Rauschen* tritt bei allen Übertragungen auf; es ist eine allgemeine Störgröße, die die Übertragung zufällig und weitestgehend gleichmäßig beeinflussen kann. Bei *Kurzzeitstörungen* kann es sich, z. B. bei CDs, um Kratzer auf diesen handeln. Hier sind mehrere Bits hintereinander fehlerhaft. Weiterhin treten solche Fehler unregelmäßig auf, da sie keiner stochastischen Größe zugrunde liegen. *Signalverformungen* treten durch Verzerrungen auf, also Störungen im Übertragungskanal, welche das Signal dämpfen oder strecken. Bei *Nebensprechen* wird, wie der Name schon suggeriert, das Signal durch Überlagerungen von anderen Übertragungen beeinflusst.

Es gibt verschiedene Arten und Weisen, eine solche Kontrollziffer zu berechnen. In dieser Unterrichtseinheit werden die vorkommenden Einsen in jeder Binärzahl gezählt. Handelt es sich dabei um eine gerade Anzahl, so wird die Prüfziffer 0 gewählt, ist die Anzahl der Einsen jedoch ungerade, so ist die Kontrollziffer 1. Diese Kontrollziffer wird auch *Paritätsbit* genannt. So wie die Prüfziffern gewählt sind, handelt es sich hierbei um eine gerade Parität, da die Ziffern so ergänzt werden, dass die übertragene Folge immer eine gerade Anzahl an Einsen hat.

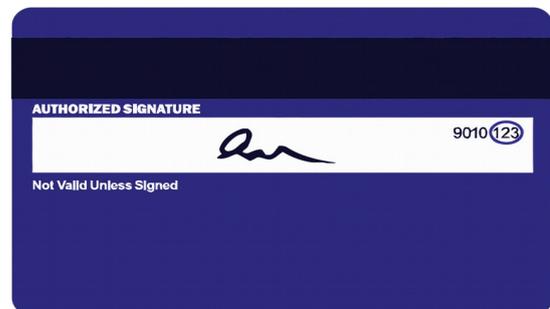


Abbildung 23: Kreditkarte mit Prüfziffer 123

Bevor nun einzelne Codes decodiert werden, sollte zunächst überprüft werden, ob die Kontrollziffer mit der entsprechenden Anzahl der Einsen in einer - nun sechsstelligen - Binärzahl übereinstimmt. Ist dies nicht der Fall, so gab es einen Fehler in der Übertragung. Da die Kontrollziffer in diesem Verfahren jedoch nur zwei verschiedene Werte haben kann, ist eine richtige Ziffer keine Garantie für eine fehlerfreie Übertragung. (Werden nur zwei statt vier Einsen übertragen, so ist deren Anzahl immer noch gerade und die Prüfziffer die Gleiche.) Deshalb gibt es deutlich kompliziertere Verfahren,

um sicherere Kontrollziffern zu bilden. Diese sichereren Verfahren bieten allerdings auch keine hundertprozentige Sicherheit.

Solche sichereren Kontrollziffern bzw. Prüfziffern finden sich beispielsweise auf Kreditkarten (die Prüfziffern sind dabei auf der Rückseite angegeben), Ausweisen, in der IBAN (die beiden Ziffern hinter dem Länderkürzel, zum Beispiel: DE21 570...) in Barcodes/QR-Codes und vielem mehr.



Abbildung 24: GTIN-Barcode, die 0 am Ende bildet die Prüfziffer



## Zusatzinformation

Prüfziffern, wie sie auf Kreditkarten, Ausweisen und in dem in Abbildung 24 dargestellten GTIN-Barcode zu finden sind, werden gebildet, indem die Zahlen des vorangehenden Codes mit festgelegten Faktoren multipliziert und dann summiert werden. Entweder bildet eine Ziffer dieser Summe dann die Prüfziffer oder die Summe wird noch anderweitig (z. B. mit Modulo) verrechnet.

Beim abgebildeten GTIN-Barcode ist die letzte Ziffer (hier die 0) die Prüfziffer. Diese wird errechnet, indem die vorangehenden Zahlen abwechselnd mit 1 und 3 multipliziert, anschließend summiert und zweimal modulo 10 gerechnet werden, also:

$$5 \cdot 1 + 0 \cdot 3 + 1 \cdot 1 + 2 \cdot 3 + 3 \cdot 1 + 4 \cdot 3 + \dots = 5 + 0 + 1 + 6 + 3 + 12 + \dots = 90 \text{ mod } 10 = 0 \text{ mod } 10 = 0$$

Die resultierende 0 stimmt mit der Prüfziffer 0 überein, somit ist dieser Barcode korrekt.

Bei den Ausweisnummern verhält es sich ähnlich, dort werden die Zahlen abwechselnd mit 7, 3 und 1 multipliziert, die Ergebnisse summiert und die letzte Ziffer der Summe bildet die Prüfziffer.

Solche Prüfsummen sind allerdings nur bei der Fehlererkennung hilfreich und können keine Fehler beheben.

Andere Verfahren, die bei der Datenübertragung auch Fehler korrigieren können, sind *Hamming-Codes*, die *Blocksicherung*, welche sich aus dem zweidimensional genutzten Paritätsbit bilden lässt und auch die *Zyklische Redundanz Prüfung (CRC für cyclic redundancy check)*.

Eine simple Art und Weise Hamming-Codes zu generieren ist, zu den genutzten Bits eine bestimmte Anzahl zusätzlicher Prüfbits einzufügen, sodass jedes zusätzliche Prüfbit für einen bestimmten Bereich eine gerade Parität herstellt.

Die eingefügten Bits kommen an Stellen, die eine Potenz von 2 sind, somit 1, 2, 4, 8, 16, 32 usw. Angenommen die Bitfolge 0110 1001 110 soll mit einem Hamming-Code gesichert werden, dann können 4 Paritätsbits (P1, P2, P3 und P4) eingefügt werden: P1 P2 0 P3 1 1 0 P4 1 0 0 1 1 1 0.

Paritätsbit	Geprüfte Stellen	Geprüfte Bits	Bestimmter Wert
P1	1, 3, 5, 7, 9, 11, 13, 15	P1, 0, 1, 0, 1, 0, 1, 0	P1 = 1
P2	2, 3, 6, 7, 10, 11, 14, 15	P2, 0, 1, 0, 0, 0, 1, 0	P2 = 0
P3	4, 5, 6, 7, 12, 13, 14, 15	P3, 1, 1, 0, 1, 1, 1, 0	P3 = 1
P4	8, 9, 10, 11, 12, 13, 14, 15	P4, 1, 0, 0, 1, 1, 1, 0	P4 = 0

Generierter Hamming-Code: 1 0 0 1 1 1 0 0 1 0 0 1 1 1 0 .

Tritt nun ein Fehler auf, kann man ihn durch die vier Paritätsbits eindeutig zuordnen und korrigieren. Bei zwei Fehlern wird das Codewort fehlerhaft korrigiert und bei drei Fehlern wird das übertragene Codewort gegebenenfalls als korrekt erkannt. Da mit dem Hamming-Code übertragene Sequenzen immer in valide Codewörter korrigiert werden können, ist er ein perfekter Code. Wegen des Aufwandes beim Verschicken zusätzlicher Bits werden Codes, bei denen die Anzahl von Kontroll- und Datenbits sehr ähnlich ist, in der Realität eher nicht verwendet. Der hier generierte Hamming-Code wird als (15, 11) bezeichnet, gängiger ist der (63, 57). Dabei gibt die erste Zahl die Anzahl der Nachrichtenbits (hier insgesamt 15 Bits) und die zweite Zahl die Anzahl der Datenbits (hier 11 Bits) an.

Ein alternatives Verfahren, welches nicht auf Prüfung der Parität beruht, ist CRC. Es basiert auf Polynomdivision und wird meist zur Datenübertragung per Ethernet und zur Datenspeicherung auf Festplatten genutzt. Allerdings ist dieses anfällig dafür, fehlerhafte Daten fälschlicherweise als korrekt zu prüfen.



Für CRC muss sich im Vorfeld auf ein Generatorpolynom geeinigt werden. Angenommen es wird das Generatorpolynom 1011 benutzt, was äquivalent zu  $1 \cdot x^3 + 0 \cdot x^2 + 1 \cdot x^1 + 1 \cdot x^0$  ist. Somit hat das Polynom Grad 3.

Beim Übertragen von Datenbits werden diese von hinten mit n Nullen verlängert, wobei n dem Grad des Generatorpolynoms entspricht. Laufen die Nutzdaten beispielsweise 100110, dann werden diese zu 100110000 ergänzt, was dem Polynom  $1 \cdot x^8 + 0 \cdot x^7 + 0 \cdot x^6 + 1 \cdot x^5 + 1 \cdot x^4 + 0 \cdot x^3 + 0 \cdot x^2 + 0 \cdot x^1 + 0 \cdot x^0$  entspricht.

Nun wird das Polynom der verlängerten Nutzdaten per Polynomdivision durch das Generatorpolynom dividiert.

$$\begin{array}{r}
 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \\
 1 \ 0 \ 1 \ 1 \\
 \hline
 - \\
 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \\
 \phantom{0 \ 0 \ 1 \ 0 \ 1} 1 \ 0 \ 1 \ 1 \\
 \hline
 - \\
 \phantom{0 \ 0 \ 1 \ 0 \ 1} 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\
 \phantom{\phantom{0 \ 0 \ 1 \ 0 \ 1} 0 \ 0 \ 0 \ 1} 1 \ 0 \ 1 \ 1 \\
 \hline
 - \\
 \phantom{\phantom{\phantom{0 \ 0 \ 1 \ 0 \ 1} 0 \ 0 \ 0 \ 1} 0 \ 0 \ 0 \ 1} 0 \ 0 \ 1 \ 1
 \end{array}$$

Die letzten drei Stellen vom Rest (hier 011) ersetzen dann die angehängten Nullen in den Nutzdaten und werden zur Überprüfung der übertragenen Nutzdaten verwendet. Die zu übertragenen Daten lauten insgesamt 100110011.

Der Empfänger dividiert seine erhaltenen Daten erneut durch das Generatorpolynom. Ist der Rest Null, wurden die Daten korrekt übertragen. Ist er ungleich Null, so wurden die Daten verfälscht.

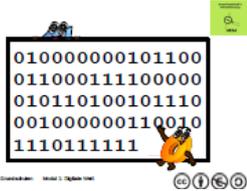
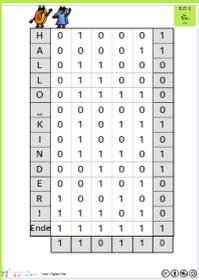
Mit dem CRC-Verfahren ist es möglich, einen Bit-Fehler zu erkennen und somit zu korrigieren. Jedoch kann es auch passieren, dass fehlerhafte Daten nicht erkannt werden, falls das Fehlerpolynom ein Vielfaches vom Generatorpolynom ist oder Fehler sowohl im Datenteil als auch im angehängten Rest auftreten.

Die oben erklärten Verfahren werden, dadurch dass sie auch Fehler korrigieren können, eher für die Datenübertragung genutzt als simple Prüfziffern. Sie sind auch insofern sicherer, da sie nicht immer auf denselben Verfahren zum Generieren beruhen wie z. B. die Prüfziffer beim Ausweis.

Diese Verfahren dienen dazu, Nachrichten weniger fehleranfällig zu übertragen, sodass der Empfänger automatisch fehlerhafte Übertragungen herausfiltern und neu anfordern kann. Dies erleichtert erheblich die Kommunikation zwischen verschiedenen Medien. Weiterhin dienen die Prüfziffern dazu, Fälschungen von Identitäten zu verhindern, sodass die Kommunikation soweit wie möglich auf Empfänger und Sender beschränkt wird.

Internetseite mit vielen verschiedenen Verfahren zur Prüfzifferberechnung: [www.arndt-bruenner.de/mathe/scripts/pruefziffern.htm](http://www.arndt-bruenner.de/mathe/scripts/pruefziffern.htm)

## Empfohlene Vorgehensweise für UE3: Fehlererkennung

Beschreibung des Unterrichtsgeschehens	Materialien
<ul style="list-style-type: none"> <li>Hängen Sie als <u>Einstieg</u> in die Unterrichtsstunde wiederum die Codetabelle mit Buchstaben und Satzzeichen an die Tafel (<i>Kopiervorlage UE2.1</i>). Neben die Codetabelle hängen Sie die mit Nullen und Einsen codierte Botschaft „Fehler“ (<i>Kopiervorlage UE3.1</i>). Übersetzen Sie mit den Kindern im Unterrichtsgespräch den Code und notieren Sie die decodierten Buchstaben an der Tafel. Damit wird die Äquivalenz zwischen <i>an/aus</i> und <i>0/1</i> deutlich.</li> </ul>	 <p>Kopiervorlage UE3.1</p>
<ul style="list-style-type: none"> <li>Wenn das Prinzip der Codierung und Decodierung den Schülerinnen und Schülern bewusst ist, nutzen Sie noch einmal die Minicodetabelle (<i>Kopiervorlage UE2.5</i>) und eine Botschaftskarte mit eingebautem Fehler (<i>Kopiervorlage UE3.2</i>). <u>Empfohlene Sozialform:</u> Partnerarbeit oder Kleingruppe mit max. drei Kindern <u>Erklärung:</u> Die Botschaft muss kurz und eindeutig sein. Ein fehlerhafter Vokal ist auch von rechtschreibschwachen Schülerinnen und Schülern leicht zu erkennen. Die Vokale haben nach vorgegebener Codetabelle folgende Codes: A 00001, E 00101, I 01001, O 01111, U 10101. Gezielt wird ein Vokal falsch übertragen, der in der Codetabelle mit nur einer Stelle abweicht. Leicht zu vertauschen sind somit die Vokale A und E.</li> <li>Die von den Kindern zu übersetzende Botschaft soll sein: „HALLO KINDAR!“</li> <li>Die Kinder übersetzen anhand der Codetabelle in der Gruppe die Botschaft und beschreiben den Fehler.</li> <li>Erarbeiten Sie mit den Schülerinnen und Schülern im Plenum die <i>eine</i> falsche Ziffer.</li> <li>Erklären Sie, dass solche Fehler bei der Übertragung von Daten tatsächlich vorkommen können.</li> <li>Besprechen Sie das Problem einer fehlerhaften Datenübertragung am Beispiel einer Robotersteuerung, einem ferngesteuerten Auto oder einer anderen Maschinensteuerung. Zur Veranschaulichung kann auch ein Kind mit den Befehlen <i>vor</i>, <i>zurück</i>, <i>rechts drehen</i> und <i>links drehen</i> durch die Klasse dirigiert werden und mit einem fehlerhaften Befehl vor Tafel, Regal oder Wand laufen.</li> </ul>	 <p>Kopiervorlage UE3.2</p>
<ul style="list-style-type: none"> <li>Im weiteren Verlauf des Unterrichts sollen die Kinder die begrenzte Möglichkeit einer Fehlererkennung durch die Übertragung einer zusätzlichen Kontrollziffer kennenlernen.</li> <li>Hängen Sie dazu das ursprüngliche „HALLO KINDER!“ an die Tafel (<i>Kopiervorlage UE3.3</i>). Die Unterrichtsvorlage enthält schon die zugefügte Kontrollziffer. <u>Erklärung:</u> Die Kontrollziffer erzeugt in jeder Reihe eine gerade Anzahl von Einsen. Als komplexere Alternative können Sie die Kopiervorlage 3.3b benutzen, bei der auch die Spalten ein Prüfbit haben. Dadurch kann man nicht nur die veränderten Reihen erkennen,</li> </ul>	 <p>Kopiervorlage UE3.3b</p>



## Abbildungsverzeichnis

- Figuren 0 und 1: erstellt vom InfoSphere-Team
- Modulgrafiken:
  - QR-Code: erstellt mit [www.qrcode-generator.de](http://www.qrcode-generator.de)
  - Glühbirnen, Schalter, Lupe: erstellt vom InfoSphere-Team

Abbildung 1: Modellierungskreislauf der Informatik (nach Grafik von Ludger Humbert).....	DW 07
Abbildung 2: QR-Code mit markierten Bereichen (erstellt mit <a href="http://www.qrcode-generator.de">www.qrcode-generator.de</a> , bearbeitet vom InfoSphere-Team).....	DW 12
Abbildung 3: QR-Code mit Fehler (Text mit 1213 Zeichen, Korrekturlevel H, kann noch gescannt werden, erstellt mit <a href="http://www.qr-manager.com/qr-code-generator.html">http://www.qr-manager.com/qr-code-generator.html</a> ).....	DW 13
Abbildung 4: Frame-QR-Code (Quelle: wikipedia.org, Autor: Tom Knox, CC BY-SA 3.0-Lizenz).....	DW 13
Abbildung 5: Design-QR-Code (Quelle: wikipedia.org, Autor: qrc-designer, CC BY-SA 3.0-Lizenz)..	DW 13
Abbildung 6: Micro-QR-Code (Quelle: wikipedia.org, Autor: Tom Knox, CC BY-SA 3.0-Lizenz).....	DW 13
Abbildung 7: iQR-Code (Quelle: wikipedia.org, Autor: Tom Knox, CC BY-SA 3.0-Lizenz).....	DW 13
Abbildung 8: Beispiel für eine Wortsammlung zum Begriff Informatik.....	DW 14
Abbildung 9: Tastatur (Quelle: pixabay.com, Autor: Clker-Free-Vector-Images, CC 0-Lizenz).....	DW 27
Abbildung 10: Informatiksystem (Quelle: pixabay.com, Autor: geralt, CC 0-Lizenz).....	DW 27
Abbildung 11: Bildschirm (Quelle: pixabay.com, Autor: Simon, CC 0-Lizenz).....	DW 27
Abbildung 12: Dosentelefon (Quelle: pixabay.com, Autor: 3dman_eu, CC 0-Lizenz).....	DW 29
Abbildung 13: Blaulicht (Quelle: pixabay.com, Autor: fsHH, CC 0-Lizenz).....	DW 29
Abbildung 14: Smarthome-Icons (Quelle: pixabay.com, Autor: Pixaline, CC 0-Lizenz).....	DW 29
Abbildung 15: Maus (Quelle: pixabay.com, Autor: OpenClipart-Vectors, CC 0-Lizenz).....	DW 30
Abbildung 16: Bildschirm (Quelle: pixabay.com, Autor: UnboxScience, CC 0-Lizenz).....	DW 30
Abbildung 17: Radio (Quelle: pixabay.com, Autor: Clker-Free-Vector-Images, CC 0-Lizenz).....	DW 30
Abbildung 18: Fernseher (Quelle: pixabay.com, Autor: TMSTBCH, CC 0-Lizenz).....	DW 30
Abbildung 19: Gegensprechanlage (Quelle: pixabay.com, Autor: CYFRAL, CC 0-Lizenz).....	DW 30
Abbildung 20: Taschenlampe (Quelle: pixabay.com, Autor: Clker-Free-Vector-Images, CC 0-Lizenz).....	DW 30
Abbildung 21: Morsecode (Quelle: pixabay.com, Autor: Rhey T. Snodgrass & Victor F. Camp, CC 0-Lizenz).....	DW 30
Abbildung 22: Telefonieren (Quelle: pixabay.com, Autor: 3dman_eu, CC 0-Lizenz).....	DW 30
Abbildung 23: Kreditkarte mit Prüfziffer 123 (Quelle: pixabay.com, Autor: 27707, CC 0-Lizenz).....	DW 36
Abbildung 24: GTIN-Barcode, die 0 am Ende bildet die Prüfziffer (Quelle: pixabay.com, Autor: OpenClipart-Vectors, CC 0-Lizenz).....	DW 36



**»Wie funktioniert der Roboter?«**

**Lehrerhandreichung  
zum Modul  
Roboter**

**Universität Paderborn  
Fachgruppe Didaktik der Informatik**



Ministerium für  
Schule und Bildung  
des Landes Nordrhein-Westfalen





Kathrin Müller, Anna Freund, Markus Kremer  
Universität Paderborn  
Fachgebiet Didaktik der Informatik

Stand März 2019

## Inhalt

1.	Einleitung.....	RO 02
2.	Übersicht der Unterrichtseinheit.....	RO 04
3.	Verknüpfung und Anbindung an die Richtlinien für die Grundschule und den Lehrplan Sachunterricht	RO 05
	3.1 Verknüpfung des Themas mit den Richtlinien für die Grundschule .....	RO 05
	3.2 Anbindung der Unterrichtseinheit zur Roboterprogrammierung an den Lehrplan für den Sachunterricht des Landes NRW .....	RO 05
	3.3 Verknüpfung mit dem Medienkompetenzrahmen NRW .....	RO 06
4.	Zu erwerbende Kompetenzen im Rahmen des Moduls „Wie funktioniert der Roboter“ .....	RO 07
5.	Worum geht es? – Grundlegende Sachinformationen zum Modul .....	RO 08
	5.1 Was ist ein Roboter?.....	RO 08
	5.2 Woraus besteht die Programmierung eines Roboters? .....	RO 08
	5.3 Was ist ein „Algorithmus“?.....	RO 15
	5.4 Wie nimmt ein Roboter eine Nachricht auf?.....	RO 15
	5.5 Formen der Programmierung eines Informatiksystems .....	RO 16
6.	Aufbau der Unterrichtsreihe .....	RO 18
	6.1 Fächerübergreifende Vorbereitung in Kunst: Wir bauen einen Roboter .....	RO 19
	6.2 Doppelstunde: Ein Roboter braucht präzise Befehle!.....	RO 20
	6.3 Doppelstunde: Arbeit am Forscherheft – Wir programmieren die Roboter! .....	RO 23
	6.4 Doppelstunde: Unser Roboter kann ...!.....	RO 26
	6.5 Fächerübergreifendes Arbeiten .....	RO 28
7.	Erläuterung des Forscherheftes und der jeweiligen Inhalte der Aufgaben .....	RO 29
8.	Übersicht über das benötigte Material .....	RO 36
	8.1 Puzzlestücke .....	RO 36
	8.2 Weitere Materialien .....	RO 38
9.	Quellennachweis .....	RO 39

## 1. Einleitung

In dem MSW-Projekt „Informatik an Grundschulen“ beleuchten die drei Module unterschiedliche fachlich-methodische Schwerpunkte der Informatik. Die Auswahl der Themen erfolgte unter fachdidaktischen, lernpsychologischen und allgemeinbildenden Gesichtspunkten. Für die unterrichtliche Umsetzung in der Schule ist jeweils ein zeitlicher Rahmen von ca. 6 – 8 Stunden vorgesehen. Grundlegende fachliche und fachdidaktische Hinweise zum Projekt „Informatik an Grundschulen“ sind im „Einführungsband“ nachzulesen. Zu jedem Modul gibt es einen spezifischen Themenband, der jeweils aus drei Teilen besteht: Lehrerbegleitheft, Lernmaterialien für Schüler/innen sowie Kopiervorlagen.

Das Modul „Wie funktioniert der Roboter?“ hat die für das Fach Informatik relevanten Themenbereiche „Algorithmik“ und „Programmierung“ zum Gegenstand. Zwar haben die Kinder keine direkten Erfahrungen mit Robotern und deren Einsatzgebieten in der Industrie oder Dienstleistungsbereichen. Allerdings kann davon ausgegangen werden, dass die Schüler/innen erste Vorstellungen zu Robotern aus Geschichten, Erzählungen, Bildern und Filmen sowie im Umgang mit Spielzeugen entwickelt haben. An diese begrifflich noch diffuse Präkonfiguration des Begriffs „Roboter“ der Kinder kann das Modul anknüpfen, um das Konzept „Roboter“ in der Vorstellungswelt der Schüler/innen im informatischen Sinne begrifflich weiterzuentwickeln und schrittweise zu präzisieren. Durch spielerisches Ausprobieren und Experimentieren können sich die Kinder mittels eines didaktischen „unplugged“ Ansatzes, zunächst ganz ohne den Einsatz von Informatiksystemen (Computern, Robotern), erste Schritte der Programmierung erschließen. Auf diese Weise lernen sie im Rollenspiel und mit selbst gebastelten Robotern Algorithmik und Programmierung als wichtige Konzepte der Informatik kennen und verstehen. Am Beispiel des Roboters wird für sie erfahrbar, dass konkrete und präzise Handlungsanweisungen zur Steuerung benötigt werden, daher ein Übergang von der oft unpräzisen Alltagssprache zu einer formal und semantisch eindeutigen „Steuerungssprache“ erforderlich ist.

In diesem Lehrerbegleitheft zum Modul „Wie funktioniert ein Roboter?“ werden wesentliche fachliche und fachdidaktische Aspekte zur Unterrichtsgestaltung und zum Einsatz der Unterrichtsmaterialien bis hin zu den Bastelanleitungen für die Papierroboter erläutert. Auf fachlicher Ebene befasst sich das Lehrerbegleitheft u.a. mit den Fragen „Was ist ein Roboter?“, „Woraus besteht die Programmierung eines Roboters?“, „Was ist ein Algorithmus?“ und „Wie nimmt ein Roboter Information auf?“ Ferner werden unter didaktischer Perspektive Anknüpfungspunkte des Moduls zu den Richtlinien für die Grundschule in NRW, zum Sachunterricht und zum fächerübergreifenden Lernen mit Bezügen zum Kunstunterricht erläutert. Weitere Ansatzpunkte zum fächerübergreifenden Lernen in Kooperation mit den Fächern Mathematik, Deutsch, Musik und Sport sind im Einführungsband dargestellt.

Das Modul folgt einem kompetenzbasierten fachdidaktischen Konzept das Kompetenzen als Einheit von kognitiven (fachlichen) und nicht-kognitiven (motivationalen, volitionalen) Komponenten versteht. In dem Projekt „Informatik an Grundschulen“ werden die mit dem Informatikunterricht verbundenen Kompetenzerwartungen mit einem Modell beschrieben, das sich an den Empfehlungen der Gesellschaft für Informatik (GI) zu den „Kompetenzen für informatische Bildung im Primarbereich“ und einer Expertise des „Haus der kleinen Forscher“ (HdkF) zur frühen informatischen Bildung orientiert. Eine genauere Darstellung der Konzepte und zur fachdidaktischen Einordnung der Module befindet sich im Einführungsband zum Projekt.

Entsprechend dieser Empfehlungen werden auch im Projekt „Informatik an der Grundschule“ die mit den Modulen verbundenen Kompetenzerwartungen Kompetenzfeldern zugeordnet, die sich jeweils durch eine Prozess- und eine Inhaltskomponente beschreiben lassen. Danach werden beobachtbare Handlungen der Schülerinnen und Schüler (Prozesse) einem fachlichen Themenbereich zugeordnet (Inhaltsbereich). Eine detaillierte Darstellung des Konzepts findet sich im Einführungsband. Für das Modul „Wie funktioniert ein Roboter?“ erläutert die unten stehende Tabelle in Form einer „Heatmap“ die entsprechenden Zuordnungen. Dabei wird deutlich, dass sich die in diesem Lehrerbegleitheft später genauer erläuterten und einzelnen Unterrichtsstunden zugeordneten Kompetenzerwartungen R-K1... R-K7 nicht eindeutig einem Kompetenzfeld zuordnen lassen, sondern oft mehrere von ihnen adressieren. Mit der hier abgebildeten Heatmap soll daher verdeutlicht werden, welche Kompetenzfelder in dem Modul schwerpunktmäßig angesprochen werden (dunklere Farbe) welche ebenfalls von Bedeutung sind (hellere Farbe) und welche in diesem Modul lediglich ergänzende Funktionen besitzen (weiß). Wie sich die mit dem in diesem Modul beschriebenen Informatikunterricht letztlich praktisch umsetzen lassen und welcher Schwerpunkt hauptsächlich fokussiert wird, liegt in der gestalterischen Freiheit der einzelnen Lehrkraft. Die Heatmap soll hier nur einen Orientierungsrahmen liefern.

Inhaltsbereiche Prozessbereiche	(I1) Information & Daten	(I2) Algorithmen & Programmierung	(I3) Sprachen & Automation	(I4) Informatik- systeme	(I5) Informatik, Mensch & Gesellschaft
(P0) Interagieren & Explorieren					
(P1) Modellieren & Implementieren	Abläufe erstellen R-K2; R-K6	Handlungs- abläufe formulieren R-K3; R-K4, R-K6	Präzise Befehle als Eingabe zur Steuerung geben R-K5		
(P2) Begründen & Bewerten					Erfahrungen mit Robotern reflektieren R-K9
(P3) Strukturieren & Vernetzen		Verbindung zu Alltagsvor- stellungen R-K8			
(P4) Kommunizieren & Kooperieren		Handlungs- vorschriften erklären R-K1, R-K4, R-K6			
(P5) Darstellen & Interpretieren		Interpretieren Handlungs- vorschriften R-K5; R-K3		Ordnen von Bestandteilen des Roboters R-K7	

## 2. Übersicht der Unterrichtseinheit

Diese Unterrichtseinheit besteht aus drei thematisch zusammenhängenden Doppelstunden, von denen die 2. und 3. Doppelstunde aufgrund der Arbeit am Forscherheft eine inhaltliche Einheit bilden.

Die erste Doppelstunde wird gekennzeichnet vom Vorwissen der Kinder zum Thema „Roboter“ sowie einer Sensibilisierung für die Bedeutung genauer und konkreter Befehle.

Der Schwerpunkt der beiden folgenden Doppelstunden liegt beim Entwickeln kleiner Algorithmen zur Steuerung eines Papp-Roboters über ein Spielfeld.

### **Intention der Lernsequenz**

Die Schülerinnen und Schüler (SuS) lernen die Funktionsweise von Robotern kennen. Sie vollziehen nach, dass ein Roboter konkrete Handlungsanweisungen braucht und fertigen selbst einfache Handlungsanweisungen zur Steuerung an.

### **Dauer**

3 Doppelstunden (jeweils 90 Minuten)

### **Materialien**

- Ggf. DVD: „Was ist was. Computer und Roboter“
- Forscherheft
- Kopiervorlagen aus dem Materialpaket. Diese werden auf Seite 36 im Lehrerbegleitheft erklärt.
- Material zum Basteln der Roboter (siehe Seite 18)

### 3. Verknüpfung und Anbindung an die Richtlinien für die Grundschule und den Lehrplan Sachunterricht

#### 3.1 Verknüpfung des Themas mit den Richtlinien für die Grundschule

„Die elektronischen Informations- und Kommunikationstechnologien sind ebenso wie die traditionellen Medien Hilfsmittel des Lernens und Gegenstand des Unterrichts. [...] Indem die Medien selbst zum Gegenstand der Arbeit im Unterricht werden, erfahren die Schülerinnen und Schüler Möglichkeiten und Beschränkungen einer durch Medien geprägten Lebenswirklichkeit.“ (MSW 2008a, S.15).

Dies ist die Begründung und Grundlegung des Unterrichtsvorhabens „Wie funktioniert der Roboter? Erste Schritte in die Programmierung“. Das Vorhaben führt die Schülerinnen und Schüler in einen Bereich des informatischen Denkens ein und ermöglicht es ihnen, sich aktiv mit dem Medium auseinander zu setzen. Durch die aktive Auseinandersetzung wird der sichere und verantwortungsbewusste Umgang eingeübt und angebahnt (vgl. §2 Abs. 4 und 5 SchulG) (MSW 2008a, S.11).

Sowohl durch die thematische Auswahl, als auch durch die didaktische Aufarbeitung des Unterrichtsvorhabens, werden grundlegende allgemeine und inhaltsbezogene Kompetenzen entwickelt, welche dem Aufbau einer Wissensbasis und der Anbahnung von Schlüsselqualifikationen dienen (MSW 2008a, S.12). Durch die Konfrontation mit der Lernaufgabe reaktivieren die Schülerinnen und Schüler ihr Vorwissen, welches sie zu Beginn der Einheit verbalisieren. Die Analyse der bisherigen Erkenntnisse und das Reflektieren über die bereits gemachten Erfahrungen mit dem Thema Roboter ermöglichen allen Kindern einen Zugang zur Aufgabe. Während des Fortschreitens des Unterrichtsvorhabens sind die Kinder immer wieder durch die Ausgestaltung der Aufgaben im Forscherheft angehalten, sowohl ihre Ideen zu strukturieren und darzustellen, als auch ihre Ergebnisse zu transferieren und anzuwenden (MSW 2008a, S.13).

Durch die fächerverbindenden Elemente bietet das Unterrichtsvorhaben auch die Möglichkeit, das kognitive Lernen mit künstlerischem, sprachlichem und sozialen Lernen zu verknüpfen und so für den Lernweg der Schülerinnen und Schüler besonders wirksam zu werden (MSW 2008a, S.12).

#### 3.2 Anbindung der Unterrichtseinheit zur Roboterprogrammierung an den Lehrplan für den Sachunterricht des Landes NRW

Laut Lehrplan hat der Sachunterricht als Teil des Bildungs- und Erziehungsauftrags die Aufgabe, die Schülerinnen und Schüler dabei zu unterstützen sich in ihrer Lebenswelt zurechtzufinden, sie zu erschließen, sie zu verstehen und sie verantwortungsbewusst mit zu gestalten (MSW 2008B, S. 39). Hier setzt die Einheit zur Roboterprogrammierung an. Während Schülerinnen und Schüler im Bereich der Nutzung der informationstechnischen Medien oft bereits im Einschulungsalter sehr erfahren sind, bietet die Einheit zur Roboterprogrammierung zusätzlich die Möglichkeit, sich diesen Teil ihrer Lebenswirklichkeit selbst zu gestalten.

Die Aufgabe des Sachunterrichts ist es, die intensive Auseinandersetzung mit wissenschaftlichen und technischen Inhalten von Beruf und Arbeitswelt, zu forcieren (MSW 2008b, S.39). Dies unterstützt die

Unterrichtseinheit, da die Schülerinnen und Schüler durch die gegebenen Aufgabenstellungen dazu angeregt werden, Fragen zu formulieren und diese zu erforschen.

Der Unterrichtsgegenstand findet sich im Bereich „Technik und Arbeitswelt“ (MSW 2008b, S.41). Das Ziel dieses Bereichs ist es, den Schülerinnen und Schülern die Bedeutung menschlicher Arbeit durch die Erkundung von Arbeitsbedingungen und –situationen näher zu bringen (MSW 2008b, S.44f.).

### 3.3 Verknüpfung mit dem Medienkompetenzrahmen NRW

Im 2017 aktualisierten Medienkompetenzrahmen Medienpass NRW wurde der Bereich „Problemlösen und Modellieren“ hinzugefügt. In diesem neu geschaffenen Kompetenzbereich ist das vorliegende Modul „Wie funktioniert ein Roboter?“ verortet.

#### 6.1: Prinzipien der digitalen Welt

Die Kinder sollen grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen, verstehen und bewusst nutzen (Medienberatung NRW 2018).

Im Modul beschäftigen sich die Kinder mit dem grundlegenden EVA-Prinzip. (Eingabe, Verarbeitung, Ausgabe).

#### 6.2: Algorithmen erkennen

Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen, nachvollziehen und reflektieren (Medienberatung NRW 2018).

Zur Vorbereitung auf das selbstständige Anfertigen von Algorithmen zum Steuern des Roboters besprechen die Kinder vorhandene Algorithmen und testen diese aus. Auch im späteren Verlauf des Moduls müssen die Kinder wieder Algorithmen nachvollziehen und reflektieren, da das selbst anfertigen im späteren Verlauf des Moduls zu hohe Ansprüche an die Kinder stellt.

#### 6.3: Modellieren und Programmieren

Probleme formalisiert beschreiben, Problemlösestrategien entwickeln und dazu eine strukturierte, algorithmische Sequenz planen. Diese auch durch Programmierung umsetzen und die gefundene Lösungsstrategie beurteilen (Medienberatung NRW 2018).

Dies ist der Schwerpunkt des Moduls. Die Kinder entwickeln Lösungen für Probleme, die sich dem Roboter stellen und modellieren einen algorithmischen Lösungsweg. Im Nachgang des Moduls können die Kinder auch an einen Computer wechseln und ihr erworbenes Wissen z.B. mit der kindgerechten Programmierumgebung Scratch weiter anwenden und vertiefen.

#### 6.1: Bedeutung von Algorithmen

Einflüsse von Algorithmen und Auswirkung der Automatisierung von Prozessen in der digitalen Welt beschreiben und reflektieren (Medienberatung NRW 2018).

Diese Kompetenz steht nicht im Mittelpunkt des Moduls, lässt sich allerdings am Ende der Einheit (im Rückgriff auf die Erfahrungen und das Wissen der Kinder aus der 1. Stunde) beachten.

## 4. Zu erwerbende Kompetenzen im Rahmen des Moduls „Wie funktioniert der Roboter“

Die Schülerinnen und Schüler . . .

**(R-K1)** erklären, dass ein Roboter präzise Befehle als Eingabe benötigt.

**(R-K2)** erstellen Abläufe zur Steuerung eines Roboters mithilfe vorgegebener Befehle auf Karten oder Bausteinen.

**(R-K3)** benennen und formulieren präzise Handlungsvorschriften.

**(R-K4)** erklären gelesene Handlungsvorschriften und -abläufe für die Steuerung eines Roboters.

**(R-K5)** interpretieren Handlungsvorschriften und -abläufe korrekt und führen sie schrittweise richtig aus.

**(R-K6)** formulieren Fragen zur Steuerung eines Roboters.

**(R-K7)** ordnen Bestandteile eines Roboters der Eingabe, der Verarbeitung und der Ausgabe zu.

**(R-K8)** erläutern Verbindungen zwischen den Themen der Unterrichtseinheit Robotik und ihren Alltagsvorstellungen.

## 5. Worum geht es? – Grundlegende Sachinformationen zum Modul

Hintergrund für diese Unterrichtseinheit bilden verschiedene Fragestellungen:

- Was sind Roboter?
- Woraus besteht die Programmierung eines Roboters? (Allgemeiner: woraus besteht ein Computerprogramm?)
- Was ist ein Algorithmus?
- Wie nimmt ein Roboter eine Nachricht auf?

### 5.1 Was ist ein Roboter?

Der Begriff „Roboter“ taucht zum ersten Mal im Theaterstück „Rossum’s Universal Robots“ von Karel Capek auf. In diesem, im Jahr 1921 geschriebenen, Stück geht es um eine Substanz, die zum Herstellen von Robotern verwendet wird. Schon in diesem Stück wird sehr früh eine oft benutzte dramaturgische Entwicklung, die Angst vor Robotern, weil sie die Menschen vernichten wollen, vorweggenommen. Ursprünglich sollten Roboter in dem Stück allerdings als „Arbeiter“ dienen – dies geht auch aus dem Begriff „Roboter“ hervor, welcher aus dem Slawischen stammt und mit „arbeiten“ übersetzt werden kann.

Allgemein stellt man sich unter einem „Roboter“ oft eine menschenähnliche Maschine vor, die Aufgaben übernehmen kann. Eine allgemeine Definition für den „Roboter“ gibt es allerdings nicht. Das Robot Institute of America definiert Roboter als „*A reprogrammable, multifunctional manipulator designed to move material, parts, tools, or specialized devices through various programmed functions for the performance of a variety of tasks.*“ (Tzvetkova 2013) (Auf Deutsch: Eine **programmierbare**, flexibel einsetzbare Maschine, gebaut um Material, Teile, Werkzeuge oder Spezialgeräte zu transportieren, welche dank unterschiedlicherer **Programmierungen** für die unterschiedlichsten Aufgabenbereiche eingesetzt werden kann.)

Die Hervorhebungen in der deutschen Übersetzung zeigen schon, dass im Kern der Definition steht, dass ein Roboter über eine Programmierung verfügen muss, die ihn erst zum Arbeiten befähigt.

Programmierbare Maschinen, also Roboter im weiteren Sinne, gibt es schon länger. Der berühmteste und einflussreichste Vorfahr ist wahrscheinlich der „programmierbare Webstuhl“ von J.M. Jacquard von 1805, welcher über eine flexibel auswechselbare Lochkarten-Programmierung beliebig komplexe Muster einfach herstellen konnte und als Grundstein für die Automatisierung von Arbeitsprozessen ein wichtiger Meilenstein in der industriellen Revolution war.

### 5.2 Woraus besteht die Programmierung eines Roboters?

Das Thema „Programmierung“ ist ein weites Feld, welches ganze Bibliotheks-Abteilungen füllt. An dieser Stelle soll es daher nur einen kurzen Überblick geben.

Damit ein Computer (und der Roboter aus dieser Unterrichtseinheit ist grundsätzlich auch nur ein Computer auf Beinen) eine Aufgabe lösen kann, braucht er Anweisungen: das Computerprogramm.

Dieses Computerprogramm muss erstellt werden, bevor der Computer die Aufgabe lösen kann. Dabei muss man als Programmierer alle Eventualitäten (Eingaben, Ausgaben, Hindernisse, mögliche Entscheidungen, ...) vorher planen und in den Befehlen des Computerprogramms abfangen.

Wie erstellt man nun so ein Computerprogramm?

Ein Computerprogramm wird in einer Programmiersprache geschrieben. Hier gibt es inzwischen zahlreiche unterschiedliche Programmiersprachen. Grob lassen sich diese Programmiersprachen unterscheiden in solche, die sehr hardwarenah, also sehr abhängig vom jeweiligen Prozessor des Systems, sind und solchen mit einem höheren Abstraktionsgrad. Hier spielt es während der Programmierung in der Regel keine Rolle welchen Prozessor das System, für welches das Programm entwickelt wird, besitzt. Hardwarenahe Sprachen werden als Maschinensprachen bezeichnet. Sprachen mit einem höheren Abstraktionsgrad werden in der Regel als Höhere Programmiersprachen bezeichnet. Darüber hinaus existieren Visuelle Programmiersprachen, sie stellen manchmal eine Art Aufsatz auf eine andere Programmiersprache z.B. dem Bereich der höheren Programmiersprachen dar. Abbildung 1 gibt einen kurzen Überblick über einen Auszug der Typen von Programmiersprachen und ihren Abstraktionsgrad.

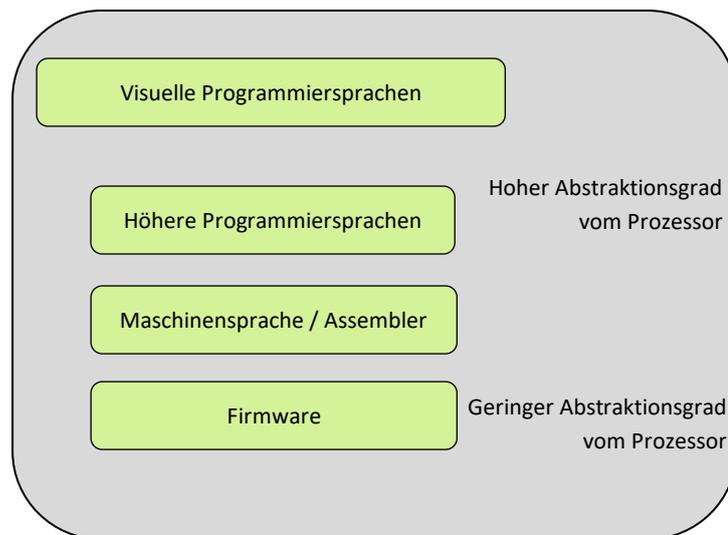


Abbildung 1 Typen von Programmiersprachen und ihr Abstraktionsgrad vom Prozessor

### **Maschinensprache:**

Der Computer selbst benötigt Anweisungen, die nur aus den Zeichen „1“ (An) und „0“ (Aus) bestehen. (Siehe hierzu auch die Ausführungen zum Material zum Binärsystem, welches im Projekt „Informatik an Grundschulen“ ebenfalls zur Verfügung steht. Das Erstellen der Befehle in Form dieser Maschinensprache wäre allerdings sehr abstrakt, kompliziert und nicht benutzerfreundlich.)

Zur Vereinfachung nutzte man ursprünglich z.B. eine Computersprache wie Assembler, die sich noch sehr nah an der Maschinensprache orientierte.

### **Höhere Programmiersprachen:**

Um das Programmieren benutzerfreundlicher zu machen wurden mit der Zeit sogenannte „Höhere Programmiersprachen“ entwickelt. In diesen Programmiersprachen werden feststehende Befehle und Zeichen als abstrakte Anweisungen benutzt, die anschließend (von einem sogenannten Compiler) in die Maschinensprache des Computers umgesetzt werden.

### **Visuelle Programmiersprachen:**

Der nächste Schritt um Programmiersprachen benutzerfreundlich zu machen, sind „Visuelle Programmiersprachen“. Diese Programmiersprachen eignen sich auch sehr gut für Kinder und Jugendliche, da sie sehr anschaulich sind. Ein Beispiel für eine solche Programmiersprache wäre „Scratch“ (siehe „mögliche Weiterarbeit“).

In dieser Programmiersprache nutzt man, ähnlich wie in dieser Einheit, „Puzzlestücke“ mit Befehlen, welche man aneinanderlegen kann. Die Gestaltung eines Computerprogramms wird immer einfacher und Programmieren ist heute keine „Hexerei“ mehr. Auch die Programmierbefehle in dieser Unterrichtsreihe orientieren sich an visuellen Programmiersprachen, um einen anschließenden, möglichen Umstieg auf den Computer zu vereinfachen.

### **Programmstrukturen**

Dennoch gibt es auch heute beim Programmieren noch einiges zu beachten.

### **Beginn und Ende**

Computerprogramme haben einen Beginn und ein Ende. Diese beiden Punkte im Programmablauf werden in den Programmiersprachen auf unterschiedlichste Weisen dargestellt. Manchmal reichen

sogar nur zwei geschweifte Klammern ( { } ), um den Start und das Ende zu definieren. Sie sind allerdings wichtig, damit der Computer am Ende des Programms mit der Arbeit aufhört und nicht seinen Arbeitsspeicher sequenziell weiter im Hinblick auf Befehle interpretiert.

*In der vorliegenden Unterrichtsreihe werden Anfang und Ende durch die Befehlskärtchen mit der Aufschrift „Beginn“ und „Ende“ repräsentiert.*

### Lineare Befehlsabfolge

Die Befehle eines Computerprogramms werden linear nacheinander abgearbeitet. Der Computer fängt mit der ersten Anweisung nach dem Anfang an und endet bei der letzten Anweisung.

*Dieser lineare Ablauf wird in der vorliegenden Unterrichtsreihe durch die ineinandergreifenden Puzzlestücke repräsentiert.*

### Befehle / Funktionen

Man unterscheidet einen einfachen Befehl und eine Funktion. Ein Befehl ist genau eine Anweisung, die ausgeführt wird. Ein Befehl könnte so z.B. sein: „Addiere zur Zahl 4 die Zahl 3 hinzu.“ (Meistens wird dieser Befehl durch eine entsprechende Rechenaufgabe dargestellt.)

```
durchschnitt()  
{  
    ergebnis1 = 3 + 4  
    ergebnis2 = ergebnis1 : 2  
}
```

Eine Funktion ist eine Art „Container“ für einen oder mehrere Befehle, die zusammengehören. Wenn man z.B. den Durchschnitt zweier Zahlen ausrechnen möchte, muss man 1. die beiden Zahlen addieren und 2. das Ergebnis durch 2 teilen. Sicherlich kann man diese beiden Befehle einfach untereinander in den „Quelltext“ schreiben. Wenn man aber öfter den Durchschnitt zweier Zahlen ausrechnen möchte, macht es Sinn, diese Befehle in eine Funktion auszulagern. Diese Funktion kann man im Laufe des Programms immer wieder aufrufen.

*Die Kärtchen „GeheVor(1)“, „DreheLinks()“, „DreheRechts()“ in der vorliegenden Unterrichtsreihe sind daher eigentlich keine Befehle sondern Funktionen, da sich hinter der Anweisung „GeheVor(1)“ für einen Roboter eine Reihe exakter Befehle („Fuß heben, Fuß nach vorne bewegen, Fuß senken, ...) verbergen.*

### Parameter

Unter einem Parameter versteht man einen Wert, den man an eine Funktion übergeben kann. Dies sorgt dafür, dass eine Funktion nicht immer mit den gleichen Angaben arbeitet, sondern den oder die Werte bei der Programmausführung verwendet, die als Parameter an die Funktion übergeben wurden.

```
durchschnitt(zahl 1, zahl 2)
{
    ergebnis1 = zahl1 + zahl2
    ergebnis 2 = ergebnis1 / 2
}
```

Dies wäre die oben schon einmal genannte Funktion, mit der ein Durchschnitt errechnet wird. Allerdings können nun 2 flexible Zahlen für die Berechnung herangezogen werden.

In einem Computerprogramm würde man diese Funktion nun z.B. folgendermaßen starten:

```
durchschnitt(3, 4)
```

Es würde demnach nun der Durchschnitt der beiden Zahlen 3 und 4 ausgerechnet.

*Grundsätzlich wird im Rahmen des Moduls beim Befehl „geheVor(1)“ bereits ein Parameter verwendet. Die Funktion geheVor kann beispielsweise wie folgt aussehen:*

```
geheVor(zahl)
{
    zaehler=0
    wiederhole
    {
        geheVor
        zaehler=zaehler+1
    }bis (zaehler==zahl)
}
```

*Diese Funktion sorgt dafür, dass der Befehl „geheVor“ so oft ausgeführt wird, wie mit dem Parameter angegeben. Im Beispiel geheVor(1) also genau einmal. (Die genaue Beschreibung der Wiederholung finden Sie im folgenden Abschnitt zu den Kontrollstrukturen).*

*Im Rahmen des Forscherheftes kommen Parameter für die SuS ab Aufgabe 6 zum Einsatz. Dort wird die Anzahl der zur Verfügung stehenden Puzzlestücke reduziert, so dass sich die Kinder (mit Hilfe eines Tippzettels) überlegen müssen, wie sie dennoch ihr Ziel erreichen können. Ans Ziel kommen sie, indem sie den „geheVor( )“-Befehl mit Bleistift (bzw. bei einlaminierten Puzzleteilen mit Folienschreiber) um den Wert für den Parameter ergänzen. „geheVor(3)“ bedeutet z.B. dass der Befehl „geheVor“ dreimal nacheinander ausgeführt wird.*

## Kontrollstrukturen

Kontrollstrukturen sind Anweisungen in Programmiersprachen, die den Ablauf des Computerprogramms flexibel steuern können. Zu den Kontrollstrukturen gehören Bedingungen („Wenn – dann – sonst“) oder Schleifen („Wiederhole bis“).

### Wenn – dann – sonst

Computerprogramme müssen oft Entscheidungen treffen. Damit dies möglich ist, nutzt man beim Programmieren eine sogenannte „Bedingung“.

```
wenn(Bedingung)
{
    Anweisung 1
}
sonst
{
    Anweisung 2
}
```

Trifft die Bedingung der bedingten Anweisung zu, so wird die Anweisung 1 ausgeführt. Trifft die Bedingung nicht zu, so wird die Anweisung 2 ausgeführt.

*Im Forscherheft muss der Roboter vor einem Hindernis entscheiden können, wie er weiterlaufen wird. Steht er vor einer Mauer, dreht er sich nach rechts. Bei anderen Hindernissen, z.B. einem Zaun, dreht er sich nach links.*

```
wenn (Mauer)
{
    dreheRechts()
}
else
{
    dreheLinks()
}
```

### Wiederhole – bis

Eine Schleife steuert, wie oft eine Gruppe von Anweisungen in einem Computerprogramm wiederholt wird. Es gibt verschiedene Schleifenformen, ihnen ist gemein, dass eine Abbruchbedingung benötigt wird. Diese Abbruchbedingung beschreibt die Voraussetzung, zu der die Schleife beendet wird. Ansonsten würde die Schleife nie enden und es würde zu einer sogenannten Endlosschleife kommen.

```
wiederhole
{
    Anweisung 1
    Anweisung 2
} bis (Abbruchbedingung)
```

*Im Forscherheft wird die Wiederholung benutzt, um die „geheVor“-Anweisung flexibel einzusetzen. (Flexibler als dies mit einem Parameter bei „geheVor“ möglich wäre.) Abbruchbedingung ist bei Verwendung der Schleife jeweils „Hindernis im Weg“. Die Anweisung innerhalb der Schleife wäre „geheVor“.*

```
wiederhole
{
    geheVor
} bis (Hindernis im Weg)
```

*Eine der Reflexionsfragen im Unterrichtsentwurf lautet „Was hat die Schleife mit einem Parameter zu tun?“. Beim „GeheVor“-Befehl handelt es sich (siehe oben) streng genommen um eine Aneinanderreihung mehrerer Befehle, um eine Funktion. Wenn der Befehl nun mit einem Parameter aufgerufen wird, werden die Befehle innerhalb dieser Funktion in einer Schleife aufgerufen, die erst abbricht, wenn die Anzahl der Durchläufe mit dem Parameter übereinstimmt.*

```
wiederhole
{
    geheVor
} bis (Parameter erreicht)
```

### 5.3 Was ist ein „Algorithmus“?

Exakt beschriebene Kochrezepte, die schriftliche Division, und Computerprogramme haben eine Gemeinsamkeit. Es sind alles Algorithmen, die eine Handlung beschreiben, bzw. als Handlungsanweisung dienen. Diese Handlungsanweisungen müssen genau sein und zum Ziel führen.

Beim Programmieren muss man eine entsprechende Handlungsanweisung für einen Computer schreiben, alle Eventualitäten berücksichtigen und passende genaue Befehle geben. Dieses strukturierte Vorüberlegen, welches im Rahmen dieser Einheit gefördert werden soll, nennt man auch „Algorithmisches Denken“.

### 5.4 Wie nimmt ein Roboter eine Nachricht auf?

Neben dem algorithmischen Denken sollen die Kinder im Laufe der Einheit das „EVA-Prinzip“ kennenlernen. Darunter versteht man ein Konzept der Datenverarbeitung, welches die Reihenfolge beschreibt, in der eine Information verarbeitet wird.

#### **E**ingabe **V**erarbeitung **A**usgabe

**Eingabe:** die Maschine wird mit einer Information versorgt („gefüttert“). Beim Computer könnte dies über eine Tastatur geschehen. Beim Roboter kann die Information über Sensoren aufgenommen werden.

**Verarbeitung:** die aufgenommene Information wird intern entsprechend der bestehenden Programmierung verarbeitet.

**Ausgabe:** die Ergebnisse der Informationsverarbeitung werden dargestellt. Dies kann z.B. über einen Monitor oder bei einem Roboter über Sprachausgabe, Gesten oder Bewegungen, etc. geschehen.

Um eine solche Information nach dem oben geschriebenen Prinzip verarbeiten zu können, muss sie zunächst in eine geeignete Repräsentationsform, d.h. in nach bestimmten Prinzipien strukturierte Daten, gebracht werden. Daten und ihre zugehörige Struktur sind somit die Repräsentation von Information. Sie werden erst durch die Interpretation zu einer Information. Genauere Informationen zum Thema Daten enthält das Modul „Digitale Welt“. Datenstrukturen sind dabei z.B. natürliche Zahlen, Dezimalzahlen, einzelne Zeichen oder Zeichenketten. Zugehörige Daten wären: natürliche Zahl -> 3; Dezimalzahl -> 3.7; einzelnes Zeichen -> „b“; Zeichenkette -> „Roboter“.

## 5.5 Formen der Programmierung eines Informatiksystems

Ein Roboter ist ein Informatiksystem mit Möglichkeiten zur Eingabe von Daten, zu deren Verarbeitung und der Ausgabe von Daten. Dazu benötigt er Sensoren, um Eingabedaten zu empfangen und Aktoren, um empfangenen Befehle z.B. in Bewegungen umzusetzen. Von anderen Informatiksystemen, die uns im Alltag häufig begegnen, ist uns dieses Konzept gut bekannt. Fernsehgerät, DVD-Rekorder, Mikrowelle, Wasch- und Spülmaschine oder Bankautomat und Ampelanlage sind hierfür nur einige Beispiele.

### Programmieren als Direkte Manipulation

Bei den meisten der vorgenannten Beispiele erfolgt die Programmierung durch direkte Manipulation, in dem wir am Gerät oder auf der Fernbedienung bestimmte Knöpfe drücken. Auch einen Roboter kann man so programmieren, indem man ihm einen Befehl übermittelt, den er dann unmittelbar ausführt, ähnlich wie z.B. bei der Fernsteuerung eines Spielzeugautos. Im Unterricht kann man das mit den Schülerinnen und Schülern spielerisch umsetzen, in dem man einem Kind, das die Rolle eines Roboters spielt, Befehle zuruft (z.B. „Gehe einen Schritt vor“), die dann unmittelbar umgesetzt werden.

### Programmieren als Abstraktion über die Zeit

Oft ist es aber so, dass zunächst ein komplettes Programm geschrieben wird, das man dem Roboter anschließend übermittelt. Wird das Programm dann gestartet, führt es der Roboter aus. Die Programmausführung erfolgt also zu einem späteren Zeitpunkt als die Entwicklung des Programms. Beim Programmieren, muss man daher vorher überlegen, welche Befehle der Roboter abarbeiten soll, damit er die gewünschten Aktionen ausführt und sicher zum Ziel gelangt. Das ist ein Szenario, das wir auch im Unterricht zunächst vorfinden. Der Roboter soll sich von einem Startfeld zu einem Zielfeld bewegen und dabei möglicherweise auch Hindernissen ausweichen. Beim Entwickeln eines Programms für derartige Operationen haben wir als außenstehende Beobachter meist einen vollen Überblick über die Größe des Feldes, die Lage der Hindernisse, Startfeld und Blickrichtung des Roboters sowie die Position des Zielfeldes, zu dem er gelangen soll. Mit den Schülerinnen und Schülern werden wir im Unterricht hauptsächlich dieses Szenario umsetzen, da man dabei einen guten Überblick über elementare Algorithmen und Kontrollstrukturen des Programmierens erhält.

### Programmieren als Abstraktion über Situationen

Wie ist es aber, wenn sich die Lage der Hindernisse auf dem „Spielfeld“ ändert, Start- und Zielposition des Roboters beliebig veränderbar sind, sogar die Größe des gesamten Spielfeldes unbekannt ist und sich ggf. sogar vorher unbekannte Objekte auf einzelnen Feldern des Spielfeldes befinden? Dann muss man beim Programmieren von einem gegebenen konkreten Szenario mit gesicherten Lagedaten von Objekten (z.B. Zaun, Mauer, Blume..) und des Roboters abstrahieren und ein Programm entwickeln, das dem Roboter „Verhaltensregeln“ für alle möglichen unterschiedlichen Zustände auf dem Spielfeld vermittelt.

### Programmieren als Wechsel der Betrachtungsperspektive

Um diese Programmieraufgabe zu bewältigen, hilft ein Wechsel der Sichtweise auf das Szenario: weg von einem außenstehenden Betrachter, der über die vollständige Information zur Größe des Spielfeldes und die Lage der Objekte auf den Feldern des Spielfeldes verfügt, hin zur beschränkten

Sichtweise aus der Perspektive des Roboters, dessen Information unvollständig, lokal begrenzt und von den Wahrnehmungsfähigkeiten seiner Sensoren abhängig ist.

Dazu benötigen wir zunächst ein einfaches konzeptuelles Modell der Wahrnehmungs- und Aktionsfähigkeiten unseres Roboters (Maschinenmodell). Ein solches Modell könnte sein: Roboter verfügt über einen Wahrnehmungssensor (z.B. Berührungssensor, Lichtsensor, Kamera..), der ihm anzeigt, welches Objekt sich auf dem Feld in seiner Blickrichtung vor ihm befindet. Als Aktoren stehen ihm Drehung (links, rechts) sowie Bewegung in Blickrichtung (vorwärts) zur Verfügung. Nun kann man zunächst ein Programm entwickeln, das den Roboter zu Bewegungen auf dem Feld befähigt, indem er Hindernissen ausweicht und durch Probieren von Bewegungen zu einem Ziel findet (z. B. einer Blume). Dieses Programm muss vollständig entwickelt sein, bevor es dem Roboter übergeben wird (Abstraktion über die Zeit).

### Roboter als autonomes System

Ein Roboter, der über ein derartiges korrekt entwickeltes Programm verfügt, das ihn in einem gegebenen Szenario (hier Spielfeld) nach Start des Programms seine Aufgabe „autonom“ erledigen lässt, kann als autonomes Informatiksystem angesehen werden. Im Alltag finden wir solche Systeme z.B. zunehmend in der Automobilproduktion (autonomes Fahren). Bei „autonomen Fahrzeugen“ werden allerdings viele unterschiedliche Sensorwerte parallel ausgewertet und gleichzeitig auch parallel unterschiedliche Aktoren gesteuert.

Im Unterricht kann man einen „autonomen Roboter“ auf dem Spielfeld simulieren, indem man zunächst ein Programm entwickelt und dieses dann einem Schüler / einer Schülerin in der Funktion eines „Roboters“ z.B. in Form von Befehlskarten übergibt. Der „Roboter“ muss seine Aktionen auf dem Spielfeld nun entsprechend seinem Programm durchführen. Die Lerngruppe kann dann sehen, ob das Programm den Roboter zur Erledigung der ihm zugeordneten Aufgabe befähigt. Dieses Konzept sollte aber nur auf einfache Szenarien angewendet werden und ist als optionale didaktische Vertiefung der Fragestellung für fortgeschrittene Lerngruppen anzusehen.

## 6. Aufbau der Unterrichtsreihe

Thema der Einheit	Kurzbeschreibung	Zeit (ca.)	Seite
<i>Vorbereitung</i> <b>„Wir bauen uns einen Roboter!“</b>	Die SuS konstruieren sich aus einfachem Material einen Roboter für die Arbeit am Forscherheft.	1 Doppelstunde	19
<i>Einführung</i> <b>„Ein Roboter braucht präzise Befehle!“</b>	Die SuS aktualisieren ihre Vorkenntnisse zum Thema „Roboter“ und erarbeiten handlungsorientiert, wie sinnvolle Befehle aufgebaut sind.	1 Doppelstunde	20
<i>Arbeit am Forscherheft</i> <b>„Wir programmieren die Roboter!“</b>	Die SuS lernen grundlegende Konstrukte der Programmierung kennen und wenden diese zunehmend gezielter an.	1 Doppelstunde	23
<i>Vertiefung</i> <b>„Unser Roboter kann ...!“</b>	Die SuS gestalten und präsentieren komplexere Algorithmen und überprüfen diese Kriterien geleitet.	1 Doppelstunde	26

## 6.1 Fächerübergreifende Vorbereitung in Kunst: Wir bauen einen Roboter

Vorbereitung: Wir bauen uns einen Roboter!	
Ziele und Kompetenzen	<p><i>Anforderungsbereich „Reproduzieren“ (AB I). Das Lösen der Aufgabe erfordert Grundwissen und das Ausführen von Routinetätigkeiten.</i></p> <ul style="list-style-type: none"> <li>• Die SuS aktivieren ihr Vorwissen in Bezug auf das Thema.</li> <li>• Die SuS gestalten einen eigenen Roboter.</li> </ul> <p><i>Anforderungsbereich „Zusammenhänge herstellen“ (AB II). Das Lösen der Aufgabe erfordert das Erkennen und Nutzen von Zusammenhängen.</i></p> <ul style="list-style-type: none"> <li>• Die SuS erstellen aus den bereit gestellten Materialien einen Roboter.</li> </ul> <p><i>Anforderungsbereich „Verallgemeinern und Reflektieren“ (AB III). Das Lösen der Aufgabe erfordert komplexe Tätigkeiten wie Strukturieren, Entwickeln von Strategien, Beurteilen und Verallgemeinern.</i></p> <p>---</p>
Zeit	⌚ 1 Doppelstunde
Materialüberblick	<ul style="list-style-type: none"> <li>📄 1 Klopapierrolle (pro Roboter).</li> <li>📄 Buntes Tonpapier (10 * 15 cm). Dieses Tonpapier wird für den Rumpf des Roboters um die Klopapierrolle gewickelt.</li> <li>📄 4 Streifen buntes Tonpapier (Breite 2 cm) zum Herstellen zweier Hexentreppen für die Arme. Länge der Tonpapier-Streifen nach Bedarf.</li> <li>📄 Bunter Tonkarton für den Kopf (Größe nach Bedarf, ca. 7 * 7 cm).</li> <li>📄 Büroklammern, Klämmerchen, Draht, etc. für die Antennen.</li> <li>📄 Alufolie sowie weitere Dekorationen für den Rumpf.</li> </ul>
Zusätzliches Material	📄 - / -
Vorbereitung	<ul style="list-style-type: none"> <li>• Zuschneiden des Tonpapiers</li> </ul>

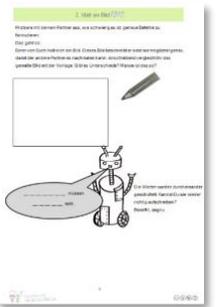
Im Kunstunterricht soll (als Vorlauf vor der ersten Doppelstunde oder auch im Anschluss daran) ein Klopapier-Rollen-Roboter gebastelt werden, mit dem die Kinder in der 2. und 3. Doppelstunde die Programme auf dem Spielfeld testen. Die Roboter können wie auf dem untenstehenden Foto aussehen. Der Fantasie der Kinder sind keine Grenzen gesetzt.



## 6.2 Doppelstunde: Ein Roboter braucht präzise Befehle!

Einführung: Ein Roboter braucht präzise Befehle!	
Kompetenzerwartungen	<p><i>Anforderungsbereich „Reproduzieren“ (AB I). Das Lösen der Aufgabe erfordert Grundwissen und das Ausführen von Routinetätigkeiten.</i></p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• ...aktivieren ihr Vorwissen in Bezug auf das Thema.</li> <li>• <b>...(R-K1)</b> erklären, dass ein Roboter präzise Befehle als Eingabe benötigt.</li> </ul> <p><i>Anforderungsbereich „Zusammenhänge herstellen“ (AB II). Das Lösen der Aufgabe erfordert das Erkennen und Nutzen von Zusammenhängen.</i></p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• <b>...(R-K3)</b> benennen und formulieren präzise Handlungsvorschriften.</li> </ul> <p><i>Anforderungsbereich „Verallgemeinern und Reflektieren“ (AB III). Das Lösen der Aufgabe erfordert komplexe Tätigkeiten wie Strukturieren, Entwickeln von Strategien, Beurteilen und Verallgemeinern.</i></p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• <b>...(R-K6)</b> formulieren Fragen zur Steuerung eines Roboters.</li> <li>• <b>...(R-K8)</b> erläutern Verbindungen zwischen den Themen der Unterrichtseinheit Robotik und ihren Alltagsvorstellungen.</li> </ul>
Zeit	⌚ 1 Doppelstunde
Materialüberblick	<ul style="list-style-type: none"> <li>📄 KV 1 aus dem Forscherheft</li> <li>📄 KV 2 aus dem Forscherheft</li> <li>📄 Vorlagenkarten zum Beschreiben der Bilder (zu KV 2)</li> <li>📄 Notizzettel (je 3 Blätter pro Kind)</li> </ul>
Zusätzliches Material	<ul style="list-style-type: none"> <li>📄 DVD-Player, Beamer, Fernseher, etc.</li> <li>📄 DVD „Was ist Was: Computer und Roboter“</li> <li>📄 reale Beispiele, Modelle oder Fotos für Roboter (Kuscheltiere, Staubsauger, Spielzeugroboter, etc.)</li> <li>📄 Ggf. Papier für den Wissenssack / Cluster</li> <li>📄 Ggf. Material für ein Roboter Gesicht</li> <li>📄 Ggf. farbige Stifte</li> <li>📄 Material zum Legen der Gefühle (Alternative Methode): 4 Strohhalme, (aneinandergebunden, für den Mund), 2 runde Augen, 2 ovale Augen, 2 Augenbrauen</li> </ul>

Handlungsschritte	Materialien / Medien
<p>Als Einstieg in die Stunde dient eine Abfrage des Vorwissens der Schüler zum Thema „Roboter“. Dies kann als einfache Sammlung an der Tafel, als Cluster mit Kreide oder Kärtchen oder in einem „Wissenssack“ erfolgen.</p>	<ul style="list-style-type: none"> <li>• Wissenssack, Tafel oder Papier für den Cluster</li> </ul>
<p>Als Hinführung zum Thema soll an dieser Stelle, wenn verfügbar, ein Ausschnitt aus der DVD „Was ist Was. Computer und Roboter“ gezeigt werden (11:48 – 20:29) In diesem Filmausschnitt werden Roboter, ihre Funktionsweise und ihre Bedeutung kurz vorgestellt.</p> <p>Die Kinder erhalten dazu einen Beobachtungsauftrag: Was sind die Vorteile von Robotern?</p> <p>Diese werden anschließend an der Tafel oder mit einem Blitzlicht gesammelt.</p> <p>Alternativ lässt sich diese Frage („Was sind die Vorteile von Robotern?“) auch ohne den Film stellen.</p>	<ul style="list-style-type: none"> <li>• DVD „Was ist Was. Computer und Roboter“</li> <li>• Beamer, DVD-Player, etc.</li> </ul>
<p>Fortsetzung der DVD (20:29 – 21:10) Es wird ein Ausschnitt von der oben genannten DVD gezeigt, bei dem ein Robotergesicht Gefühle darstellen soll. Dem Roboter wird ein entsprechender Befehl gegeben („Glücklich“) und das Gesicht stellt das Gefühl dar.</p> <p>Wenn die DVD nicht verfügbar ist, ist es auch möglich, diese Szene als Übergang zu den genauen Anweisungen zu überspringen.</p>	<ul style="list-style-type: none"> <li>• DVD „Was ist Was. Computer und Roboter“</li> <li>• Beamer, DVD-Player, etc.</li> </ul>
<p>Nach dem Beispiel von der DVD bekommen die Kinder in Partnerarbeit die Aufgabe, sich gegenseitig Gefühle zu nennen, die sie mit ihrem Gesicht (ggf. mit dem ganzen Körper) darstellen sollen. (ca. 5 Minuten)</p> <p>Alternativ: Diese Übung lässt sich auch mit Material durchführen (siehe Materialliste). Aufgabe wäre in diesem Fall das Legen der Gefühle mit dem entsprechenden Material.</p>	
<p>Im Sitzkreis machen nun einige Teams ihre „Gefühle“ vor. Die restlichen Kinder sollen das Gefühl erkennen, anschließend wird gemeinsam überlegt, was genau bei dem Gefühl passiert. (z.B. „Freude“: grinsen, „strahlendes Gesicht“, kleine Falten, Augen etwas geweitet, etc.)</p> <p>Diese einzelnen Punkte werden jeweils zu dem Gefühl notiert.</p> <p>Der Lehrer erklärt, dass ein Roboter auf die Anweisung: „Zeig ein freudiges Gesicht!“ nur reagieren kann, wenn er weiß, was er machen muss. Er benötigt genaue Anweisungen dafür.</p>	
<p>Die Kinder erhalten nun für die Überlegung, welche Befehle konkret sind, ein Arbeitsblatt mit Befehlen. Sie müssen herausfinden, welche der Befehle so genau sind, dass man sie ohne Rückfrage ausführen kann. Zusätzlich sollen sie einige der ungenauen Befehle verbessern.</p>	<ul style="list-style-type: none"> <li>• Arbeitsblatt</li> </ul>

	
<p>Anschließend sollen die Kinder mithilfe eines „Schneeball-Systems“ Kriterien für „gute Anweisungen“ finden. Schneeball-System: Zuerst überlegt sich ein Kind 3 Kriterien, dann tauscht es sich mit dem Partner aus und sie reduzieren ihre insgesamt 6 Kriterien auf 3, anschließend werden in einer Vierer-Gruppe die 6 nun vorhandenen Kriterien wiederum auf die 3 wichtigsten reduziert.</p> <p>Jede Vierer-Gruppe stellt nun die Kriterien vor, sie werden an der Tafel notiert.</p>	<ul style="list-style-type: none"> <li>• „Schmierpapier“, je 3 Blätter pro Kind</li> </ul>
<p>Zum Abschluss stellt der Lehrer die Frage, warum Anweisungen für einen Roboter genauer sein müssen als für einen Menschen. Je nach verbleibender Zeit kann diese Frage im Plenum diskutiert werden oder in kurzen Diskussionen untereinander in einer anschließenden Blitzlichtrunde besprochen werden. (Siehe didaktische Analyse: Roboter benötigen konkrete Anweisungen, da sie nicht, wie Menschen, auf Erfahrungen und Interpretationen zurückgreifen können.)</p>	
<p>Evtl. bleibt am Ende der Doppelstunde noch Zeit. Dies ist zum Beispiel möglich, wenn auf den Einsatz der DVD verzichtet wird. In dem Fall bieten sich noch weitergehende Übungen oder Aufgaben an:</p> <ul style="list-style-type: none"> <li>• Die Kinder können die Aufgabe 2 aus dem Forscherheft („Malt ein Bild“) bearbeiten, bei der sich die Kinder Rücken an Rücken setzen und sich gegenseitig ein Bild beschreiben, welches der jeweilige Partner anhand der Anweisungen nachmalen muss. Es dient eigentlich als Einstieg in das Forscherheft um zu Beginn der Doppelstunde das Thema „genaue Anweisungen“ noch einmal aufzugreifen. Als vertiefende Übung am Ende der 1. Doppelstunde kann die Aufgaben aber ebenso eingesetzt werden.</li> <li>• Die Kinder können ggf. noch einmal Verbesserungen an ihren „Klopapierroboter“ vornehmen bzw. mit ihrem Klopapierroboter beginnen.</li> </ul>	<ul style="list-style-type: none"> <li>• Arbeitsblatt</li> </ul>  <ul style="list-style-type: none"> <li>• Bildkarten</li> <li>• Material für die Klopapierroboter (siehe oben)</li> </ul>

### 6.3 Doppelstunde: Arbeit am Forscherheft – Wir programmieren die Roboter!

Arbeit am Forscherheft: Wir programmieren die Roboter	
Kompetenzerwartungen	<p>Anforderungsbereich „Reproduzieren“ (AB I). Das Lösen der Aufgabe erfordert Grundwissen und das Ausführen von Routinetätigkeiten.</p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• ...(R-K2) erstellen Abläufe zur Steuerung eines Roboters mithilfe vorgegebener Befehle auf Karten oder Bausteinen.</li> </ul> <p>Anforderungsbereich „Zusammenhänge herstellen“. (AB II) Das Lösen der Aufgabe erfordert das Erkennen und Nutzen von Zusammenhängen.</p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• ...(R-K3) benennen und formulieren präzise Handlungsvorschriften.</li> <li>• ...(R-K4) erklären gelesene Handlungsvorschriften und -abläufe für die Steuerung eines Roboters.</li> </ul> <p>Anforderungsbereich „Verallgemeinern und Reflektieren“. (AB III) Das Lösen der Aufgabe erfordert komplexe Tätigkeiten wie Strukturieren, Entwickeln von Strategien, Beurteilen und Verallgemeinern.</p> <p>Die SuS...</p> <p>...(R-K6) formulieren Fragen zur Steuerung eines Roboters.</p>
Zeit	⌚ Eine Doppelstunde
Materialüberblick	<ul style="list-style-type: none"> <li>📄 Forscherhefte für alle Schülerinnen und Schüler</li> <li>📄 Puzzlestücke in passender Anzahl (Angabe hier pro Team) <ul style="list-style-type: none"> <li>○ 1x Beginn</li> <li>○ 1x Ende</li> <li>○ 2x Wenn – dann</li> <li>○ 2x Wiederhole bis</li> <li>○ 12x geheVor(1)</li> <li>○ 6x geheVor(__)</li> <li>○ 6x dreheLinks()</li> <li>○ 6x dreheRechts()</li> <li>○ 1x stoppeRoboter</li> <li>○ 2x BlankoPuzzlestück</li> <li>○ 4x Mauer</li> <li>○ 4x Zaun</li> <li>○ 1x Blume</li> </ul> </li> <li>📄 Tippzettel für Aufgabe 6</li> <li>📄 Wenn vorhanden: selbstgebastelte Roboter</li> </ul>
Zusätzliches Material	- / -
Vorbereitung	<ul style="list-style-type: none"> <li>• Kopieren der Forscherhefte in ausreichender Zahl (1 Heft pro Kind, damit jedes Kind die Ergebnisse mitnehmen kann)</li> <li>• Kopieren der Spielfelder in ausreichender Anzahl (1 Spielfeld pro Team)</li> <li>• Kopieren und ausschneiden der Puzzlestücke in ausreichender Anzahl</li> </ul>

(1 Satz pro Team).

- Aufhängen der Tipp-Zettel

Handlungsschritte	Materialien / Medien
<p>Als Einstieg in die Stunde dient eine kurze Blitzlichtrunde im Sitzkreis, in der jedes Kind eine konkrete, genaue Anweisung geben soll.</p>	
<p>Die Lehrkraft präsentiert als „Anweisungen“ auch die Befehle „geheVor (1)“, „dreheLinks()“, „dreheRechts()“ (als Zeichnung der Puzzlestücke an der Tafel oder nur mündlich). Sie erklärt den Kindern, dass sie mit Hilfe dieser Befehle ihren Roboter passend zu den Aufgaben im Forscherheft über ein Spielfeld steuern sollen. Ggf. kann dies einmal in der Klasse mit einem Schüler/ einer Schülerin, der als „Roboter“ durch die Klasse dirigiert wird, ausprobiert werden. Dieses Szenario ermöglicht es auch einmal über die Aspekte der direkten Manipulation und der Abstraktion über Zeit und ggf. auch Raum zu sprechen. In dem zum Beispiel das Roboterkind Befehle direkt ausführt, im Sinne der direkten Manipulation, oder ein zuvor erstelltes Programm „ausführt“.</p> <p>Anschließend stellt der Lehrer im Sitzkreis das Forscherheft und das Material vor.</p> <ul style="list-style-type: none"> <li>• Die Kinder arbeiten zu zweit an den Programmieraufgaben.</li> <li>• Sie bearbeiten das Forscherheft von vorne nach hinten durch.</li> <li>• Sie sollen die Programmieraufgaben zuerst mit den Puzzle-Teilen legen, dann das Programm am Spielfeld ausprobieren und es, wenn es funktioniert, in das Forscherheft übertragen.</li> <li>• Beim Ausprobieren soll ein Kind der Reihe nach die Befehle vorlesen und das andere Kind steuert entsprechend den Roboter übers Spielfeld. An dieser Stelle sollte darauf geachtet werden, dass ein Programm vorab vollständig aufgeschrieben wird, da es sich sonst lediglich um eine direkte Manipulation und nicht um eine Abstraktion über die Zeit handelt.</li> <li>• Zu Aufgabe 6 gibt es einen Tipp. Dieser hängt im Klassenraum aus.</li> <li>• Auf der 1. Seite im Forscherheft finden die Kinder auch diese Hinweise.</li> </ul> <p>Bei Bedarf kann der Lehrer auch die einzelnen Aufgaben kurz erklären. Vorgesehen ist es aber, dass die Kinder sich die Aufgabenstellungen eigenständig erarbeiten.</p>	<ul style="list-style-type: none"> <li>• Forscherhefte</li> </ul>
<p>Die Kinder arbeiten nun mit ihrem Partner an den Aufgaben im Forscherheft. Sie beginnen mit Aufgabe 2 („Malt ein Bild!“) bzw. Aufgabe 3, falls die 2. Aufgabe schon am Ende der 1. Doppelstunde als vertiefende Übung benutzt wurde.</p>	<ul style="list-style-type: none"> <li>• Tippzettel</li> <li>• Puzzlestücke</li> <li>• Bildkärtchen (Mauern,...)</li> <li>• Bildkarten</li> </ul>
<p>Am Ende der Doppelstunde werden die Schüler nicht alle denselben Stand haben, da sie mit Sicherheit unterschiedlich schnell vorankommen. Daher bieten sich für das Ende der Doppelstunde verschiedene Abschlüsse an:</p> <ol style="list-style-type: none"> <li>1. In einer gemeinsamen Reflexionsphase am Ende kann man das Vorgehen</li> </ol>	<p>Bei Fragestellung b)</p> <ul style="list-style-type: none"> <li>• Puzzlestücke zum Programmieren</li> </ul>

<p>der Kinder beim Bearbeiten der Aufgaben besprechen. Ziel wäre bei dieser Reflexion, eine planvollere Herangehensweise der Kinder an die Aufgaben zu fördern. (Reflexionsfragen: siehe 2.)</p> <p>2. Es findet eine dezentrale Reflexion in einer Kleingruppe statt. Die nicht beteiligten Schüler arbeiten in der Zeit am Forscherheft weiter. Mögliche Fragestellungen wären:</p> <ol style="list-style-type: none"><li>Welches ist das kürzeste Programm? Warum sollte man versuchen, das Programm möglichst kurz zu halten?</li><li>In der Kleingruppe bei der dezentralen Reflexion wird von zwei Schülern/innen ein Programm gelegt. Die Gruppe reflektiert über die Herangehensweise des Teams an die Aufgabe.</li><li>Schaut euch, jetzt wo ihr die Schleife kennt, noch einmal den Parameter an. Könnten „Schleife“ und „Parameter“ etwas miteinander zu tun haben?</li><li>Testet die unterschiedlichen Spielfeldsituationen beim Einsatz der Bedingten Verzweigung. Funktioniert ein und dasselbe Programm für unterschiedliche Spielfeldsituationen?</li></ol> <p>3. Da die Arbeit am Forscherheft in der 2. und 3. Stunde einen zusammenhängenden Block bildet, wäre es auch grundsätzlich denkbar, die Kinder einfach durcharbeiten zu lassen. In diesem Fall fehlt allerdings ein Abschluss in der 2. Doppelstunde.</p>	<p>Bei Fragestellung c)</p> <ul style="list-style-type: none"><li>Kopie der Tippzettel für die Reflexion</li></ul>
--	--

**Anmerkung:** Bedingt durch den Aufbau des Forscherheftes werden die Kinder, wie schon erwähnt, nicht alle gleichzeitig an der gleichen Aufgabe arbeiten. Es bietet sich aber je nach Klasse an, dass man (wenn die ersten Gruppen die Aufgabe 5 abgeschlossen haben) eine gemeinsame Zwischenreflexion zu den bisher bekannten Befehlen einschiebt.

Lerngruppen, die nicht so frei arbeiten können oder wollen, steht es frei das Forscherheft anders zu nutzen. Hierfür kann das Material bspw. einzeln mit den SuS erarbeitet werden. Außerdem kann das Material in Form einer Lerntheke präsentiert werden, bei der die Lehrkraft notiert, welche Aufgaben für die Lerngruppe eher einfach oder eher schwierig sind.

## 6.4 Doppelstunde: Unser Roboter kann ...!

Vertiefung: Unser Roboter kann ...!	
Kompetenzerwartungen	<p><i>Anforderungsbereich „Reproduzieren“ (AB I). Das Lösen der Aufgabe erfordert Grundwissen und das Ausführen von Routinetätigkeiten.</i></p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• <b>...(R-K7)</b> ordnen Bestandteile eines Roboters der Eingabe, der Verarbeitung und der Ausgabe zu.</li> </ul> <p><i>Anforderungsbereich „Zusammenhänge herstellen“ (AB II). Das Lösen der Aufgabe erfordert das Erkennen und Nutzen von Zusammenhängen.</i></p> <p>Die SuS...</p> <ul style="list-style-type: none"> <li>• <b>...(R-K3)</b> benennen und formulieren präzise Handlungsvorschriften.</li> <li>• <b>...(R-K4)</b> erklären gelesene Handlungsvorschriften und -abläufe für die Steuerung eines Roboters.</li> <li>• <b>...(R-K5)</b> interpretieren Handlungsvorschriften und -abläufe korrekt und führen sie schrittweise richtig aus.</li> </ul> <p><i>Anforderungsbereich „Verallgemeinern und Reflektieren“ (AB III). Das Lösen der Aufgabe erfordert komplexe Tätigkeiten wie Strukturieren, Entwickeln von Strategien, Beurteilen und Verallgemeinern.</i></p> <p>Die SuS...</p> <p><b>...(R-K6)</b> formulieren Fragen zur Steuerung eines Roboters.</p>
Zeit	⌚ Eine Doppelstunde
Materialüberblick	<ul style="list-style-type: none"> <li>📄 Material der letzten Doppelstunde (falls es am Ende eingesammelt wurde)</li> <li>📄 Ggf. ein vergrößertes Spielfeld für die Tafel oder Sitzkreis</li> <li>📄 Ggf. große Kopien (passend zum großen Spielfeld) <ul style="list-style-type: none"> <li>○ 3x Mauern</li> <li>○ 3x Zäune</li> <li>○ 2x Blume</li> <li>○ 1x Roboter</li> </ul> </li> <li>📄 Ggf. etwas vergrößerte Kopien der Puzzle-Stücke zum Arbeiten im Sitzkreis oder an der Tafel</li> <li>📄 Aufhängen des Tippzettels zu Aufgabe 6</li> </ul>
Zusätzliches Material	📄 - / -

Handlungsschritte	Materialien / Medien
Der Lehrer erklärt den SuS, dass sie heute am Forscherheft weiterarbeiten werden. Zur Wiederholung steuert die Klasse ein Kind (oder mehrere Kinder) mit den bekannten Befehlen durch den Klassenraum.	
<p>Die Kinder arbeiten nun weiter an dem Forscherheft. Die Lehrperson gibt Hilfestellungen oder nutzt die Zeit und den Arbeitsfortschritt der Kinder, um mit einigen Kindern weitere dezentrale Reflexionen durchzuführen. Die Fragestellungen wären analog zu den Fragestellungen aus der 2. Doppelstunde.</p> <ol style="list-style-type: none"> <li>Welches ist das kürzeste Programm? Warum sollte man versuchen, das Programm möglichst kurz zu halten?</li> <li>In der Kleingruppe wird bei der dezentralen Reflexion von 2 Schülern/innen ein Programm gelegt. Die Gruppe reflektiert über die Herangehensweise des Teams an die Aufgabe.</li> <li>Die Gruppe überlegt sich eine alternative Erklärung zum Parameter.</li> </ol>	<ul style="list-style-type: none"> <li>• Forscherhefte</li> <li>• Tippzettel</li> <li>• Puzzlestücke</li> <li>• Bildkärtchen (Mauern,...)</li> </ul>
Anschließend arbeiten diese Teams weiter am Forscherheft und beschäftigen sich mit dem EVA-Prinzip (Eingabe/ Verarbeitung/ Ausgabe).	
<p>Am Ende der Doppelstunde (ca. 10 Minuten) können anhand von Beispiellösungen der SuS und der Aufgaben im Forscherheft die oben genannten dezentralen Reflexionsfragen im Plenum diskutiert werden. Dafür wird das große Spielfeld mit Magneten an die Tafel geheftet oder in den Sitzkreis gelegt. Die vorstellenden Teams legen / heften ihr Programm ebenfalls an die Tafel. Bevor das Programm ausgeführt wird, lässt die Lehrperson von den Kindern erklären, was passiert und stellt die Frage, ob das Programm funktionieren kann. Dann „läuft“ der Roboter, vom Programm gesteuert und von den Kindern bewegt, das Spielfeld ab. Alternativ oder ergänzend können auch neue Spielfeldsituationen eingebracht werden und gemeinsam Programme für diese Situationen entwickelt und z.B. zum Einsatz der Schleife weiterentwickelt werden. Dies können Situationen wie in Aufgabe 8 sein oder ganz neue Situationen, in denen der Roboter von einem Start zum Zielpunkt kommen muss.</p> <p>Bei der Verwendung der Bedingten Verzweigung, taucht zum ersten Mal im Ansatz eine Abstraktion über den Raum auf. Ein Programm kann für unterschiedliche Ausgangssituationen (Blume und Zaun können auf jeweils zwei unterschiedlichen Feldern liegen) genutzt werden. Daher ist es sinnvoll mit den SuS an dieser Stelle die unterschiedlichen Spielfeldsituationen in Kombination mit ein und demselben Programm zu thematisieren.</p> <p>Als abschließende Diskussion kann es hier sinnvoll sein, die Schülervorstellungen zu Robotern vom Anfang der Unterrichtseinheit nochmal aufzugreifen. Fragestellungen könnten dann z.B. sein: „Sind Roboter nun wirklich gefährlich?“ etc.</p>	<ul style="list-style-type: none"> <li>• Großes Spielfeld</li> <li>• Große Bildkärtchen (Mauern,...)</li> </ul>

## 6.5 Fächerübergreifendes Arbeiten

Fach	Mögliches fächerübergreifendes Fortführen
Kunst	Roboter zeichnen
Deutsch	Entwickeln eines Algorithmus zur Lösung eines Rechtschreibphänomens (z.B. „s/ ss/ ß“); Bofinger, M.: Graf Tüpo, Lina Tschornaja und die Anderen; (Bofinger 1991) Lornsen, B & Jakobs, G.: Robbi, Tobbi und das Fliewatüüt (Lornsen, Jakobs 2009)
Mathematik	Entwickeln eines Algorithmus für ein schriftliches Rechenverfahren
Musik	Bewegungslied „Roboter“ von Detlef Jöcker, (Jöcker 2012) Lied „Roboter“ von Kraftwerk (Kraftwerk 1978)
Sport	„Roboter-Spiel“. Ein Kind „steuert“ ein anderes, in dem es taktile „Befehle“ gibt, z.B. Klopfen auf die Schultern als Befehl für eine Drehung in die jeweilige Richtung.

## 7. Erläuterung des Forscherheftes und der jeweiligen Inhalte der Aufgaben

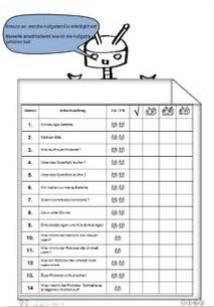
Im Folgenden werden die Seiten des Forscherheftes kurz erklärt. Kursiv geschrieben ist das Ziel, welches mit der jeweiligen Aufgabe verfolgt wird. Grau hinterlegt sind theoretische Hintergrundinformationen zum Programmieren. Die Buchstaben-Zahlen-Kombination in Klammern verweist auf die Kompetenzen, die auf Seite 4 in diesem Lehrerbegleitheft zu finden sind.

### Seite 2:



Auf Seite 2 befinden sich die Arbeitsanweisungen für die Kinder, so wie sie auch im Unterricht vorab besprochen werden sollen.

### Seite 3: Übersicht

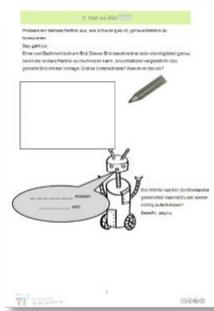


Auf Seite 3 finden die Kinder eine Übersicht, wie sie es auch vom Stationenlernen kennen. Sie sollen dort Ihren Arbeitsfortschritt eintragen und kurze Rückmeldungen zur Aufgabe geben.

### Seite 4: Eindeutige Befehle (R-K1/3)

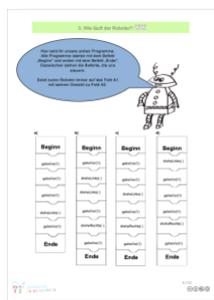


Beim Programmieren nutzt man eindeutige Befehle. Die Schüler sollen mit dieser Aufgabe für konkrete und eindeutige Anweisungen sensibilisiert werden.

**Seite 5: Malt ein Bild (R-K3)**

Beim Erstellen der kurzen Programme zum Steuern des Roboters durch das Labyrinth greifen die Kinder auf vorgegebene Anweisungen auf den Puzzlestücken zurück. Um vorab noch einmal für konkrete Anweisungen sensibilisiert zu werden, sollen sie sich gegenseitig jeweils ein Bild beschreiben und dieses jeweils anhand der Beschreibungen nachmalen. Dafür nutzen sie eine der vorgegebenen Bildkarten aus dem Arbeitsmaterial.

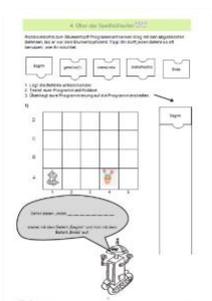
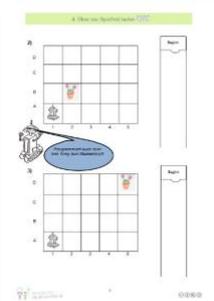
In den Lückentext sollen sie anschließend noch die Wörter „Befehle“ und „genau“ eintragen.

**Seite 6: Wie läuft euer Roboter? (R-K5)**

Nun sollen sich die Kinder mit den Befehlen und den Puzzlestücken vertraut machen. Dafür steuern sie ihren Roboter über das Spielfeld. Sie beginnen im Feld A1 und schauen (das ist wichtig für die Bewegungsrichtung beim ersten Befehl) nach rechts zum Feld A2.

Bei dieser und den folgenden Aufgaben ist es wichtig, bei den SuS ein Bewusstsein dafür zu schaffen, dass ein Roboter immer ein vollständig erstelltes Programm ausführt. (Abstraktion über Zeit anstatt direkter Manipulation)

Die Lösungen der einzelnen Aufgaben befinden sich am Ende der Seite.

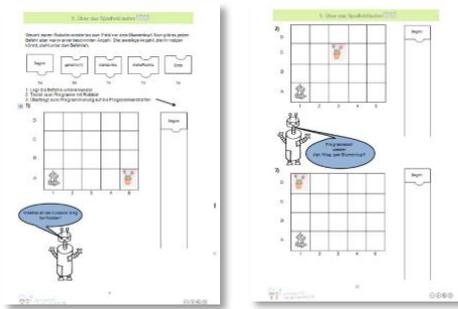
**Seite 7 / 8: Über das Spielfeld laufen 1 (R-K2/4/5)**

Die Schüler sollen bei dieser Aufgabe im Forscherheft ihren Roboter über das Feld steuern. Sie legen jeweils ein Programm mit den Puzzlestücken, um den Blumentopf zu erreichen. Die Erfahrung zeigt, dass die Kinder hier viele verschiedene Möglichkeiten ausprobieren und z.B. auch mal versuchen, ein möglichst langes Programm zu schreiben.

Zu Beginn des Programms muss der Befehl „Beginn“ stehen. Das Programm endet jeweils mit „Ende“.

*Ziel der Aufgabe: die Kinder sollen die Grundstruktur des Programms (Beginn – Ende) und die Nutzung der Befehle an einer einfachen Aufgabe üben.*

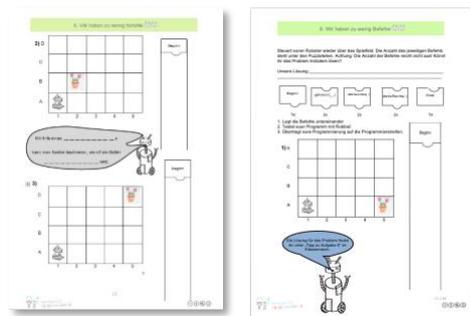
Erläuterung: bei den verschiedenen Programmiersprachen gibt es dafür unterschiedliche Lösungen. Manchmal findet man explizit den Befehl „Start“ und „Ende“, manchmal wird der Beginn eines Programms durch „{“ und das Ende durch „}“ angezeigt.

**Seite 9/10: Über das Spielfeld laufen 2 (R-K2/4/5)**

Nachdem die Kinder den grundsätzlichen Umgang mit den Befehlen geübt haben, sollen sie bei dieser Aufgabe die Programme nur mit einer bestimmten Anzahl Puzzlestücke legen können. Dies setzt voraus, dass die Kinder gezielt überlegen, mit welchen Befehlen sie das Ziel am besten erreichen können.

*Ziel der Aufgabe: die Kinder sollen eine effiziente, kurze Lösung für die Programme finden.*

Erläuterung: Computerprogramme sind in der Regel möglichst effizient und ohne unnötige ‚Umwege‘ aufgebaut, damit sie ressourcenschonend ausgeführt werden können und der Programmcode übersichtlich bleibt.

**Seite 11/12: Wir haben zu wenig Befehle! (R-K2/4/5)**

Die Kinder stoßen bei dieser Aufgabe auf die Problematik, dass ihnen Befehle zum Programmieren fehlen. Der Tippzettel erklärt ihnen, dass sie auf das Puzzlestück hinter den Befehl in Klammern eine Zahl schreiben können. Diese Zahl (man nennt sie „Parameter“) gibt an, wie oft der Befehl ausgeführt wird. Damit muss der Befehl nicht mehrmals hintereinander aufgeschrieben werden.

Die Kinder sollen auf dem Arbeitsblatt die Lösungswörter „Parameter“ und „wiederholt“ einsetzen. Diese Wörter finden die Kinder auf dem Tipp-Zettel.

*Ziel: Die Kinder sollen ihre Programme kürzen, sich Schreibarbeit abnehmen und die Programme übersichtlicher gestalten, indem sie Variablen benutzen.*

Erläuterung: Bei dieser Aufgabe werden für die SuS Parameter eingeführt (vgl S. 12).

Die Befehle, die auf den Puzzlestücken stehen, sind streng genommen keine für sich stehenden Befehle, sondern sogenannte „Methoden“ oder „Funktionen“. Methoden sind häufig vorkommende Befehlsblöcke, welche man nicht immer wieder neu schreiben möchte. Wenn eine Methode beim Programmieren aufgerufen wird, kann man Daten in Variablen speichern, die diese Methode benötigt. Mit Parametern (s.o.) werden diese Daten beim Aufruf der Methode deren Variablen zugewiesen.

Wenn die Methode „GehWasserflaschenHolen(5)“ lautet, würde die in Klammern stehende Zahl bedeuten, dass 5 Flaschen Wasser geholt werden sollen. Stünde dort eine 4, würden nur 4 Flaschen Wasser geholt.



Auch die Anwendung der „Verzweigung“ ist vergleichsweise kompliziert. Daher sollen die Kinder auch die Verzweigung nur als Konstrukt kennenlernen und nicht selber mit der bedingten Verzweigung programmieren. Stattdessen sollen sie, ähnlich wie bei Aufgabe 7, ein Programm mit einer Verzweigung durchlaufen. Durch die Variation der Position der Blume spielen die Kinder das erste Programm zweimal durch und erleben damit die Flexibilität.

Außerdem sollen sie auf der Seite 15 die Begriffe „Verzweigung“ und „Entscheidungen“ eintragen.

Mit den Kindern sollte außerdem thematisiert werden, dass an dieser Stelle ein Programm für unterschiedliche Ausgangssituationen funktioniert. Die Kinder wechseln hier zwischen ihrer Perspektive als Beobachter der Situation und der Perspektive des Roboters („Welcher Gegenstand wird vom Sensor/Roboter erkannt“).

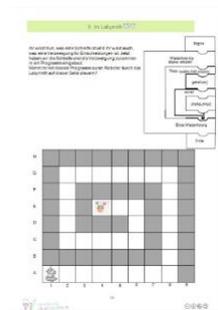
*Ziel der Aufgabe: die Kinder sollen den Befehl zum Treffen von Entscheidungen kennen lernen. Zusätzlich kommt an dieser Stelle ein Einfluss der Abstraktion über den Raum zum Tragen. Ein und dasselbe Programm funktioniert für unterschiedliche Ausgangssituationen.*

Erläuterung: sogenannte Abfragen werden in der Regel beim Programmieren durch folgende Befehlskombination dargestellt.

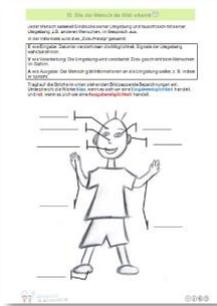
```
If(Bedingung)
{
    [Hier steht, was passiert, wenn die Bedingung zutrifft.]
}
else
{
    [Hier steht, was passiert, wenn sie nicht zutrifft.]
}
```

Analog dazu ist das Puzzlestück aufgebaut. Im Gegensatz zum Befehls-Puzzlestück, bei dem jeweils nur ein Befehl ausgeführt wird (dreheRechts oder dreheLinks) können beim richtigen Programmieren eine beliebige Anzahl an Befehlen eingefügt werden. Der „else“-Zweig kann auch komplett entfallen.

### Seite 16: Im Labyrinth (R-K7)



Die Kombination aus Schleifen und Verzweigungen machen Programme sehr flexibel. Das Programm auf Seite 16 kombiniert diese beiden Strukturen miteinander. Wie in Aufgabe 14 und 15 sollen die Kinder nicht selber eine Aufgabe legen, sondern das Programm im Labyrinth austesten.

**Seite 17: Wie der Mensch die Welt erkennt (R-K7)**

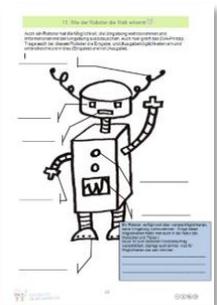
Hauptaugenmerk dieser Einheit liegt auf dem Anbahnen eines algorithmischen Denkens bei den Kindern.

Teams, die die Aufgaben zügig und erfolgreich durchlaufen, sollen sich am Ende aber noch einmal mit einem weiteren Teilbereich beschäftigen. Dem „EVA“-Prinzip. Die Abkürzung steht für „Eingabe – Verarbeitung – Ausgabe“.

Unter **Verarbeitung** versteht man alles das, was in den vergangenen Aufgaben im Mittelpunkt stand. Die Verarbeitung der Befehle, um den Roboter agieren zu lassen.

Bei Aufgabe 7 haben sich die Kinder evtl. schon selber die Frage gestellt, woran der Roboter eigentlich bemerkt, dass er vor einem Hindernis steht. Die „Erfassung“ des Hindernisses versteht man in der Robotik als **Eingabe** von Information. Wenn der Roboter Information wiedergibt, spricht man von **Ausgabe**.

Bei der Aufgabe 10 im Forscherheft sollen die Kinder sich als Vorübung überlegen, welche Körperteile des Menschen der Ein- bzw. der Ausgabe dienen.

**Seite 18: Wie der Roboter die Welt erkennt (R-K7)**

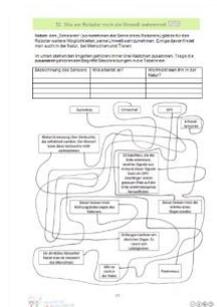
Aufgabe 11 behandelt die Fragestellung, wie ein Roboter die Umgebung wahrnehmen kann und wie er mit ihr kommunizieren kann. Dabei geht es erst einmal um leicht erkennbare Eingabe-/ Ausgabe-Möglichkeiten, welche manchmal eine Analogie zum Menschen haben.

Eingabe: „Kamera“ (optische Sensoren), Sensoren an den Händen und Füße, Schaltknöpfe, Antennen

Ausgabe: Lautsprecher (Mund), Displays, Hände, Antennen

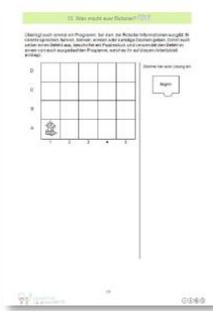
Die Kinder werden natürlich nicht die Begrifflichkeiten „Sensoren“ nutzen, sondern werden andere Begrifflichkeiten (im Zweifelsfall „Augen“) nutzen.

Die Kinder sollen sich außerdem überlegen, wie ein Roboter seine Umgebung evtl. noch wahrnehmen kann.

**Seite 19: Wie ein Roboter noch die Umwelt wahrnimmt (R-K7)**

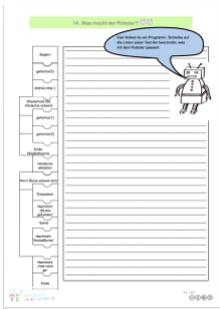
Neben den aufgrund der Analogie zum Menschen eher naheliegenden Ein- und Ausgabe-Möglichkeiten gibt es weitere Eingabe-Möglichkeiten, die ggf. Vorbilder in der Natur haben. Vier dieser Möglichkeiten (Gyroskop, Ultraschall, GPS und Infrarot-Sensoren) können die Kinder auf diesem Blatt kennenlernen.

**Seite 20: Euer Roboter soll „sprechen“ (R-K2/4/5/7)**



Basierend auf den Ausgabe-Möglichkeiten der letzten Aufgaben sollen die Kinder sich nun selber Befehle überlegen, die sie für eine Rückmeldung des Roboters nutzen können. Dabei ist der Kreativität der Kinder keine Grenze gesetzt. Der Roboter soll über das Spielfeld laufen, ggf. auf Gegenstände treffen und passende Rückmeldungen geben.

**Seite 21: Was macht der Roboter? (R-K4/5)**



Bei der letzten Aufgabe, die aber höchstwahrscheinlich nur von den schnellen Kindern erreicht wird, sollen die Kinder noch einmal einen vorgegebenen *Algorithmus interpretieren* und die Handlungen des Roboters in Form einer kurzen Geschichte aufschreiben.

**Seite 22: Ein Brief an Robbie**



Am Ende des Forscherheftes wendet sich Robbie noch einmal an die Kinder. In einem Lückentext-Brief greift er die Inhalte des Forscherheftes und die verwendeten Begriffe wieder auf. Die Begriffe stehen unterhalb des Briefes und die Kinder sollen diese in die richtigen Lücken des Textes einfügen.

## 8. Übersicht über das benötigte Material

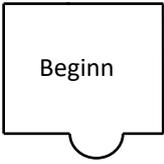
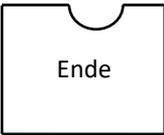
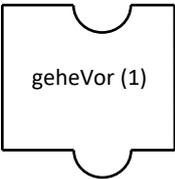
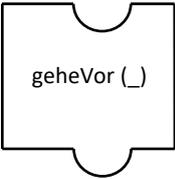
Für die Durchführung der Unterrichtsreihe muss vor Ort noch Material ausgedruckt, kopiert und zurechtgeschnitten werden. Es befindet sich in der Datei „Material.pdf“.

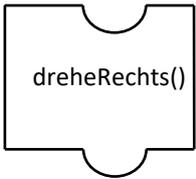
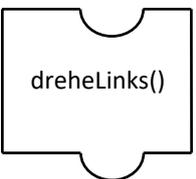
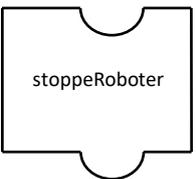
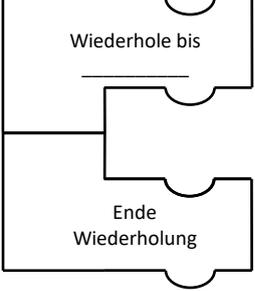
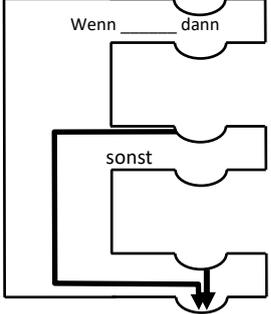
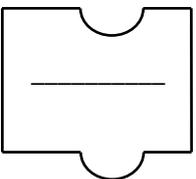
Dabei handelt es sich zum Beispiel um die „Puzzlestücke“, welche von den Kindern zum Programmieren aneinandergelegt werden.

Im Rahmen der natürlichen Differenzierung vom Kinde aus können die Puzzleteile den Kindern verpflichtend oder bei Bedarf zur Verfügung gestellt werden. Hierbei können die Puzzleteile zur Bearbeitung der Aufgabe als auch zur Kontrolle und Präsentation des eigenen Lösungswegs genutzt werden. Im Rahmen der bisherigen unterrichtlichen Arbeit im Projekt IaG hat sich gezeigt, dass die konkret handelnde Auseinandersetzung mit den Puzzleteilen vor allem bei komplexeren Aufgabenstellungen hilfreich ist.

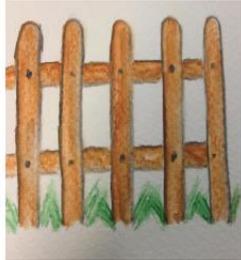
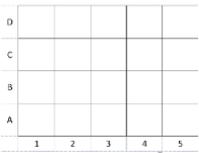
### 8.1 Puzzlestücke

Neben den hier abgebildeten Puzzlestücken, finden Sie im Material zusätzliche Puzzleteile zur Differenzierung, die eine grafische Unterstützung mit Pfeilen bieten.

Material	Bezeichnung	Beschreibung	Benötigte Anzahl pro Team
	Beginn	Mit diesem Puzzlestück beginnt jedes Programm, welches die Kinder legen werden. Es ist im Forscherheft bei jeder Aufgabe schon eingezeichnet.	1x
	Ende	Mit diesem Puzzlestück endet jedes Programm, welche es die Kinder legen werden.	1x
	geheVor (1)	Dieser Befehl „steuert“ den Roboter um ein Feld nach vorne. Bei Aufgabe 6 wird dieses Puzzlestück mit einem Bleistift oder ähnliches um die Variable ergänzt.	12x
	geheVor (_)	Dieser Befehl „steuert“ den Roboter um so viele Felder nach vorne, wie der Nutzer dies festlegt. Dies wird bei Aufgabe 6 als Parameter benötigt.	6x

	dreheRechts()	Dieser Befehl dreht den Roboter in Laufrichtung nach rechts um.	6x
	dreheLinks()	Dieser Befehl dreht den Roboter in Laufrichtung nach links um.	6x
	stoppeRoboter	Dieser Befehl stoppt den Roboter. Anschließend werden keine weiteren Befehle ausgeführt. Das Programm ist beendet. Der Befehl findet Verwendung bei Aufgabe 7 und 8, wenn das Ziel (die Blume) erreicht wurde.	1x
	Schleife	<p>Mit diesem Puzzlestück wird die Schleife (Aufgabe 8) dargestellt. Mit Bleistift wird hinter „Wiederhole bis“ eingetragen, wann die Schleife beendet werden soll.</p> <p>Der Befehl, der wiederholt wird (geheVor(), dreheLinks(), dreheRechts(), ...), wird in die Lücke gelegt.</p> <p>Man kann den Schleifenkopf (Wiederhole bis) und den Schleifenfuß (Ende Wiederholung) auch auseinanderschneiden und mehrere Befehle dazwischen legen.</p>	2x Ggf. 2 – 4 Reserve-Puzzlestücke am Pult
	Abfrage	Mit diesem Puzzlestück wird eine Abfrage abgebildet. (Siehe dazu auch die Erklärung zur Seite 13/14 im Forscherheft) Zwischen „Wenn“ und „dann“ wird auf den Strich die Bedingung eingetragen (z.B. „Mauer“). Wenn die Bedingung erfüllt ist, wird das Puzzlestück ausgeführt, welches man in die obere Lücke legen kann. Wenn die Bedingung nicht erfüllt ist, wird das Puzzlestück ausgeführt, welches in der unteren Lücke liegt.	2x
	Blanko-Puzzlestück	Dieses Puzzlestück kann bei Aufgabe 13 eingesetzt werden, um es mit einem eigenen Befehl zu beschriften.	2x

## 8.2 Weitere Materialien

Material	Bezeichnung	Beschreibung	Benötigte Anzahl pro Team
	Blumentopf	Der Blumentopf ist jeweils das Ziel, welches der Roboter erreichen soll.	1x
	Mauer	Die „Mauer“ ist eines der Hindernisse, bei denen der Roboter entscheiden muss, in welche Richtung er sich dreht.	4x
	Zaun	Die „Zaun“ ist eines der Hindernisse, bei denen der Roboter entscheiden muss, in welche Richtung er sich dreht.	4x
	Spielfeld	Im Materialpaket befindet sich weiterhin die Vorlage des Spielfeldes, das auf DIN A3-Größe vergrößert werden muss.	1x

## 9. Quellennachweis

Bofinger, M. (1991). Graf Tüpo, Lina Tschornaja und die anderen. Leipzig, Farber&Farber.

Jöcker, D.(2012). Mein nagelneuer Roboter. Detlev Jöckers 40 schönste Kindergartenlieder. Münster: Menschenkinder-Verlag (Universal Music). Text verfügbar unter: [https://www.europa-kinderwelt.de/media/files/Download-Liederbuch\\_Kindergarten\\_5663.pdf](https://www.europa-kinderwelt.de/media/files/Download-Liederbuch_Kindergarten_5663.pdf) (letzter Zugriff 08.03.2019)

Kraftwerk (1978). Roboter. Die Mensch-Maschine. Klingklang (EMI)

Lornsen, B. & Jakobs, G. (2009). *Robbi, Tobbi und das Fliewatüüt*. Stuttgart Wien: Thienemann.

Medienberatung NRW (Hrsg.), (2018). Medienkompetenzrahmen NRW. Verfügbar unter: <https://medienkompetenzrahmen.nrw.de> (letzter Zugriff 08.03.2019).

Ministerium für Schule und Weiterbildung des Landes NRW(Hrsg.),(2008),„Richtlinien für die Grundschulen des Landes NRW.

Ministerum für Schule und Weiterbildung des Landes NRW (Hrsg.), (2008). Lehrplan Sachunterricht für die Grundschulen des Landes Nordrhein-Westfalen.

Tzvetkova, G.V., (2013). The First Bulgarian Industrial Robot “PROBO” 1987. Journal of Theoretical and Applied Mechanics, 43 (4), ( S. 93-96). ISSN: 0861-6663.

### Abbildungen

Die in dem Modul „Wie funktioniert ein Roboter?“ verwendeten Abbildungen und Fotos wurden sämtlich von Projektmitarbeitern erstellt und können entsprechend der CC O Lizenz verwendet werden.







**»Ich habe ein Geheimnis!«**

**Lehrerhandreichung  
zum Modul  
Kryptologie**

**Bergische Universität Wuppertal  
Fachgebiet Didaktik der Informatik**



Ministerium für  
Schule und Bildung  
des Landes Nordrhein-Westfalen

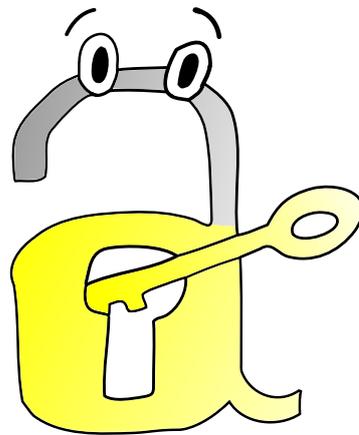


## **Federführung**

Martin Fricke & Ludger Humbert

### **unter Mitarbeit von**

Kathrin Arera, Kerstin Grah,  
Hans Haase, Kathrin Haselmeier,  
André Hilbig, Stefan Gehrman,  
Daniel Losch, Dorothee Müller,  
Philipp Rumm, Denise Schmitz  
und den Lehrkräften unserer Korrespondenzschulen



Stand: 16. März 2019

# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort zum Modul Kryptologie</b>	<b>KR 01</b>
<b>2</b>	<b>Aufbau des Moduls Kryptologie</b>	<b>KR 02</b>
<b>3</b>	<b>Das Modul Kryptologie und die Kompetenzen für informatische Bildung im Primarbereich</b>	<b>KR 03</b>
<b>4</b>	<b>Informatik an Grundschulen – Einordnung des Moduls Kryptologie</b>	<b>KR 04</b>
4.1	Informatischer Modellierungskreislauf im Modul Kryptologie . . . . .	KR 04
4.2	Die Fachgebiete der Informatik . . . . .	KR 06
4.3	Zu erwerbende Kompetenzen im Rahmen des Moduls Kryptologie . . . . .	KR 07
<b>5</b>	<b>Grundlegende Sachinformation zur Kryptologie</b>	<b>KR 08</b>
5.1	Steganographie . . . . .	KR 09
5.2	Codierung . . . . .	KR 10
5.3	Transposition . . . . .	KR 12
5.4	Substitution . . . . .	KR 14
5.5	Verschlüsselung im Alltag . . . . .	KR 15
5.6	Glossar . . . . .	KR 15
<b>6</b>	<b>Didaktische Hinweise</b>	<b>KR 19</b>
6.1	Gesprächsführung . . . . .	KR 19
6.2	Sprachsensibel im Fachunterricht . . . . .	KR 19
6.3	Allgemeine Hinweise zu Möglichkeiten der Differenzierung . . . . .	KR 19
<b>7</b>	<b>Angebote zur Unterrichtsgestaltung – Modul Kryptologie</b>	<b>KR 20</b>
	Übersicht über die Unterrichtseinheiten . . . . .	KR 21
	Einkaufsliste . . . . .	KR 23
	Einstieg: Informatik in der Lebenswirklichkeit – Im Supermarkt . . . . .	KR 24
	UE 1: Steganographie – Verstecken und Verbergen . . . . .	KR 27
	UE 2: Codierung – Der Freimaurercode . . . . .	KR 33
	UE 2a: Codierung – Codes sind überall (optional) . . . . .	KR 37
	UE 3: Transposition – Die Skytale . . . . .	KR 41
	UE 4: Substitution – Caesar-Verschlüsselung mit Code-Scheibe . . . . .	KR 47
	UE 5: Verschlüsselung im Alltag – Messenger und Co. . . . .	KR 51
<b>8</b>	<b>Weiterführende Materialien – Modul Kryptologie</b>	<b>KR 56</b>
<b>9</b>	<b>Verwendungsnachweis – Modul Kryptologie</b>	<b>KR 57</b>

# 1 Vorwort zum Modul Kryptologie

Informatik – an Grundschulen? Sollen Schülerinnen und Schüler nicht zunächst einmal Lesen, Schreiben und Rechnen lernen, bevor sie mit Informatiksystemen arbeiten?

Korrekt! Das können wir nur bestätigen.

Allerdings wird oft verkannt, dass genau diese Kompetenzen viel mehr mit Informatik zu tun haben, als man zunächst vermutet. Unser Ansatz im Projekt **Informatik an Grundschulen (IaG)** besteht darin, dass Schülerinnen und Schüler informatische Prinzipien kennen lernen, die ohne die Nutzung von Informatiksystemen einen handelnden, erfahrungsgeleiteten und erkundenden Zugang zur Informatik erlauben.

Die Wissenschaft Informatik und hier vor allem das Fachgebiet Didaktik der Informatik hat ein großes Interesse daran, dass Ideen, die theoretisch geklärt sind, in der Praxis wirksam werden. So ist es uns – als Mitarbeitende der Fachdidaktik Informatik – ein großes Anliegen, Elemente der informatischen Allgemeinbildung für Schülerinnen und Schüler in der Grundschule erfahrbar zu machen.

Unsere Expertise und unser Budget erlauben es nicht, eine Erprobung und eine Evaluation unserer Ansätze und Ideen aus eigener Kraft zu gewährleisten. Daher sind wir besonders dem Ministerium für Schule und Bildung des Landes Nordrhein-Westfalen dankbar, dass Ressourcen in Form von beauftragten Grundschullehrkräften und materielle Unterstützung zur Verfügung gestellt wurden, um Erfahrungen mit der Umsetzung zu machen.

Im Rahmen der Erprobungen sind – zunächst beginnend mit unseren Ideen, diese aber in der Folgezeit weit überschreitend – Materialien entwickelt und in der Arbeit mit Grundschülerinnen und Grundschulern erprobt worden, die hier als Handreichung dokumentiert werden, damit weitere Kolleginnen und Kollegen in die Lage versetzt werden, ebenfalls Erfahrungen mit Informatik an Grundschulen (IaG) zu machen.

Ganz herzlich bedanken für die Entwicklung, die Erprobung, die Schulung der Kooperationsschullehrkräfte und die Beiträge zur qualifizierten Überarbeitung der hier vorliegenden Materialien möchte ich mich bei Frau KATHRIN HASELMEIER, Frau KATHRIN ARERA, und bei Herrn MARTIN FRICKE, dem die Federführung bei der Kompilation der hiermit vorliegenden Lehrerhandreichung zur Kryptologie übertragen wurde. Sie haben wesentliche Arbeit im Projekt IaG am Standort Wuppertal geleistet.

Viele Stunden der Diskussion, des Abwägens und der Besprechung zur Dokumentation an deren zielführender Gestaltung maßgeblich Frau Dr. DOROTHEE MÜLLER beteiligt war, führten zu den hier in exzellenter Weise dokumentierten Erfahrungen in Form von grundlegenden Bemerkungen, Unterrichtsideen und nicht zuletzt ausgearbeiteten und er-

probten Unterrichtsmaterialien.

Für die Realisierung wird das Satzsystem  $\text{\LaTeX}$  genutzt. So entstehen qualitativ hochwertige, typografisch wertvolle Materialien. Mein Dank gilt daher Herrn PHILIPP RUMM, Herrn HANS HAASE, Frau DENISE SCHMITZ, Herrn STEFAN GEHRMANN, die als Hilfskräfte die Grundschullehrkräfte mit  $\text{\LaTeX}$  und anderen Hilfsmitteln vertraut machten und ihnen beim Bewältigen der durchaus steilen Lernkurve unterstützend zur Seite standen.

Trotz der Unterstützung konnte die schiere Vielfalt der notwendig im Zusammenhang mit  $\text{\LaTeX}$  und vor allem mit  $\text{\LaTeX}$  zu nutzenden (und damit auch zu beherrschenden) Bibliotheken nur dadurch zu der vorliegenden Qualität getrieben werden, dass mit dem Informatiklehrer Herrn ANDRÉ HILBIG sowohl fachwissenschaftliche als auch fachdidaktische Expertise in diese Materialerstellung Eingang fand. Auch ihm gebührt mein Dank, denn es ist nicht immer einfach, »mal eben« den Notensatz, die Gebärdensprache oder den EAN-Code mit Hilfe von  $\text{\LaTeX}$  zu setzen.

Die Erstellung der Materialien ist eine zentrale Voraussetzung für die weitere Arbeit – das Ziel besteht darin, dass Schülerinnen und Schüler informatische Kompetenzen entwickeln. Um dieses Ziel zu erreichen, konnten im Projektzusammenhang Kooperationschulen gewonnen werden, die den Unterrichtseinsatz der Materialien – nach vorbereitenden Schulungen – erprobten und uns hervorragende Rückmeldungen gaben.

Der standortübergreifende Austausch von Ideen, die Diskussion von Wegen zum Ziel und die fundierte Weiterentwicklung aller Materialien war nur möglich dank der intensiven Zusammenarbeit mit den Informatikfachdidaktikgruppen der universitären Standorte Paderborn und Aachen. Gerade die befruchtenden Impulse aus dem lebhaft geführten Ringen um den richtigen Weg trugen zu einer maßgeblichen Qualitätssteigerung der hiermit vorliegenden Materialien bei.

Neben den direkt im Projektzusammenhang tätigen Personen möchte ich mich bei Frau BRIGITTE SCHULTZ für die professionelle Unterstützung in allen formalen Belangen, die den Fortgang des Projekts betreffen, bedanken.

Last not least – gilt mein Dank Prof. Dr. ANDREAS FROMMER, dem Prorektor für Studium und Lehre der Bergischen Universität Wuppertal für die unterstützende Begleitung bei allen Entscheidungen der Universitätsleitung, die dieses Projekt erst ermöglicht haben.

Die Ergebnisse sind überzeugend – Danke!

LUDGER HUMBERT  
Didaktik der Informatik  
Bergische Universität Wuppertal

## 2 Aufbau des Moduls Kryptologie

In dieser Handreichung finden sich fachdidaktische Hinweise zur Umsetzung von informatischen Gegenständen – also sowohl Informatikinhalten als auch von Methoden der Informatik. Dabei werden die Phänomenbereiche der Informatik sowie der informatische Modellierungskreislauf als fachdidaktische Grundlagen dargestellt. Am Beispiel der Kryptologie wird ein inhaltliches Themenfeld aufgeschlossen. Hierbei wird zu jedem Teilbereich je ein kurzer Sachinput vorgestellt, der durch praktische Umsetzungsvorschläge und Kopiervorlagen für die Schüler- und Lehrerhand ergänzt wird. Dem Modul vorangestellt ist eine beispielhafte Einstiegsstunde zu Informatik in der Lebenswirklichkeit: *Informatik im Supermarkt*.

### Die Unterthemen im Überblick

**Beispielhafter Einstieg – Informatik in der Lebenswirklichkeit** Die Schülerinnen und Schüler besuchen einen nahe gelegenen Supermarkt und setzen die Informatikbrille auf – Welche informatischen Phänomene finden sich an diesem Lernort? Alternativ wird der Supermarkt mit Bildkarten besprochen.

Anschließend überlegen und verschriftlichen die Schülerinnen und Schüler, was sie mit dem Wortfeld Informatik assoziieren.

**Steganographie** Die Schülerinnen und Schüler finden verborgene Nachrichten und tauschen sich über den Grad der Sicherheit aus. Wenn alle Verstecke gefunden oder bekannt sind, ist die Steganographie wirkungslos.

**Codierung** Die Schülerinnen und Schüler nutzen z. B. die Freimaurer-Chiffre, indem sie mit einer Codetabelle selber codieren und decodieren. Eve als Angreiferin greift schnell die Nachricht ab, da sie auch die Codetabelle kennt – eine Verbesserung muss her.

**Transposition** Papierstreifen werden auf Holzstäbe gewickelt: die Skytale ist erfunden! Die Schülerinnen und Schüler lernen Fachbegriffe wie *Algorithmus* und *Schlüssel* und werden selbst zum Angreifer auf die Nachrichten.

**Substitution** Mit Code-Scheibe und -Tabelle verschlüsseln die Schülerinnen und Schüler wie Caesar im alten Rom. Der Angreifer hat immer mehr Mühe, den Algorithmus zu knacken. Bei einigen Verfahren ist aber auch der Schlüsseltausch unter den Eingeweihten umständlich.

**Verschlüsselung im Alltag** Zurück zum Anfang: Wo begegnen uns die nun bekannten kryptologischen Phänomene im Alltag? Die Schülerinnen und Schüler diskutieren Angriffsszenarien auf Messenger und Co.

Allgemeine Hinweise zur unterrichtlichen Umsetzung wurden in der Handreichung zur Einführung in die Informatik angegeben.

### 3 Das Modul Kryptologie und die Kompetenzen für informatische Bildung im Primarbereich

Inhaltsbereich	(1) Information und Daten	(12) Algorithmen	(13) Sprachen und Automaten	(14) Informatiksysteme	(15) Informatik, Mensch und Gesellschaft
<b>Prozessbereich</b>					
(P1) Modellieren und Implementieren	analysieren den Algorithmus zur Verschlüsselung von Nachrichten (Daten) – die Kompetenz trägt zu [I1; P5] bei	führen Algorithmen schrittweise aus ..... entwerfen einen Algorithmus zur Verschlüsselung von Nachrichten (Daten)			»knacken« verschlüsselte Nachrichten mit Hilfe eines bekannten Verfahrens – die Kompetenz trägt zu [I1; P3], [I2; P5] und zu [*; P4] bei
(P2) Begründen und Bewerten					beschreiben Analogien und Unterschiede zwischen der realen und der digitalen Welt
(P3) Strukturieren und Vernetzen		erarbeiten eine Methode zur systematischen Entschlüsselung von verschlüsselten Nachrichten (Daten) und wenden diese an – die Kompetenz trägt zu [I3; P1] und [I5; P5] bei			
(P4) Kommunizieren und Kooperieren		charakterisieren den Begriff Algorithmus		beschreiben ihren persönlichen Umgang mit Informatiksystemen	
(P5) Darstellen und Interpretieren		erläutern gegebene Algorithmen		kennen ein Verfahren zur Transposition – die Kompetenz trägt zu [I1; P1, P5] bei	beschreiben ihren persönlichen Umgang mit Diensten des Internet

(vgl. Gesellschaft für Informatik e. V., 2019)

## 4 Informatik an Grundschulen – Einordnung des Moduls Kryptologie

Das Projekt **Informatik an Grundschulen (IaG)** soll den Schülerinnen und Schülern verschiedene Facetten der Informatik verfügbar machen, und sie darin unterstützen, ein besseres Verständnis für Informatik zu entwickeln. Jedes der drei Module beleuchtet einen besonderen inhaltlich-methodischen Schwerpunkt der Informatik. Da Grundschullehrkräfte bisher in ihrer Ausbildung weder fachlich noch fachdidaktisch für die Umsetzung von Elementen der Informatik qualifiziert werden, sind aktuell Fortbildungen unabdingbar. Die Vorbereitung und Begleitung dieser Fortbildungen stellt eine der Herausforderungen dar, der wir uns mit der Vorlage dieser Handreichung stellen.

Die Autoren dieser Handreichung schlagen folgende allgemeine Definition für Informatik vor:

### Definition Informatik

Informatik ist die Wissenschaft von der systematischen und automatisierten Verarbeitung von Information.

### 4.1 Informatischer Modellierungskreislauf im Modul Kryptologie

Ein informatischer Modellierungskreislauf (vgl. Abb. 1) sowie die drei Phänomenbereiche der Informatik wurden bereits in der allgemein-einführenden Handreichung erläutert. Anbei findet sich eine auf das Modul Kryptologie zugeschnittene beispielhafte Anwendung des Modellierungskreislaufs.

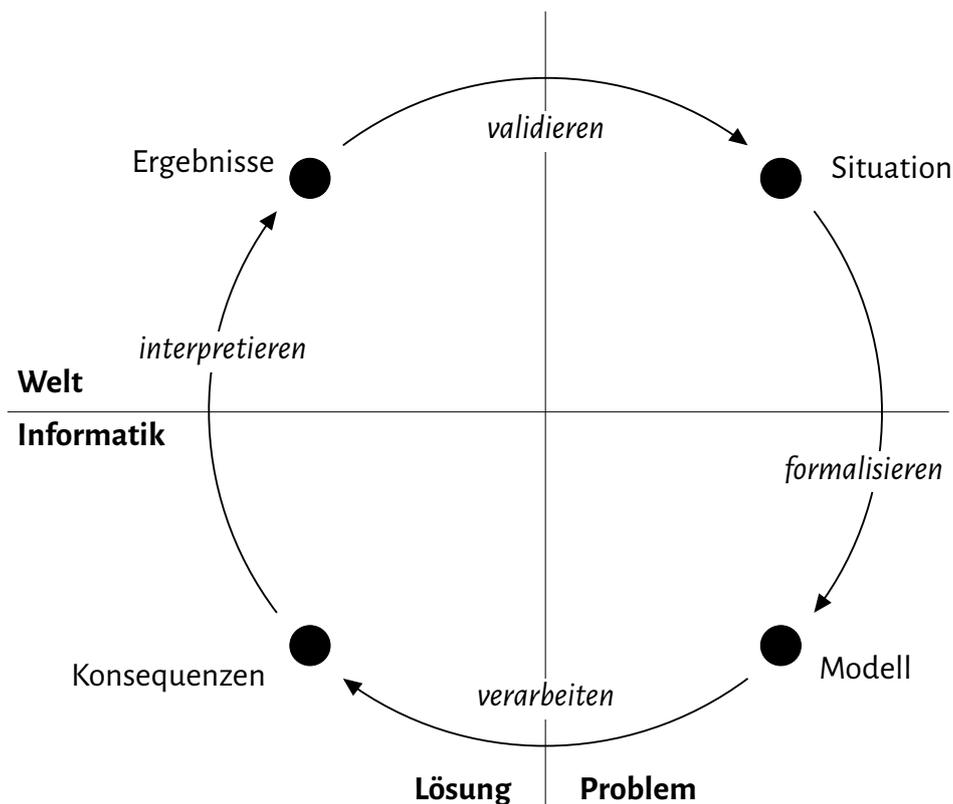


Abbildung 1: Modellierungskreis der Informatik (nach Humbert, 2006, S. 14).

### 1. Situation

Der Zustand *Situation* stellt in der wirklichen Welt eine Problemstellung dar.

**Alice und Bob möchten Nachrichten austauschen, ohne dass Eve mitlesen kann.**

*Erläuterungen zu Alice, Bob und Eve finden Sie auf Seite KR 09.*

### 2. Modell

Im Übergang zum nächsten Zustand wird das Problem *formalisiert*. Somit entsteht ein *Modell*. Ein Modell ist als gestaltetes Abbild der Realität zu verstehen und umfasst die zur Problemlösung notwendigen Elemente.

**Alice und Bob machen sich eine Skizze über den Kommunikationsweg und entwickeln eine Möglichkeit, Eve vom Lesen abzuhalten, z. B. indem sie einen Code benutzen.**

### 3. Konsequenzen

Das so entstandene Modell kann *verarbeitet* (oder *abgearbeitet*) werden. Aus der Verarbeitung ergeben sich Konsequenzen.

**Alice und Bob erstellen eine Code-Tabelle und treffen Absprachen zur Umsetzung.**

### 4. Ergebnisse

Durch *Interpretieren* der Konsequenzen gelangt man zurück in horizontale Ebene der *Welt*. Die *Ergebnisse* stellen eine *Lösung* in der Lebenswirklichkeit dar.

**Alice und Bob nutzen die Code-Tabelle, um ihre Nachrichten vor dem Zugriff Eves zu schützen. Eve schafft es aber, den Code zu knacken. Eine bessere Methode muss her. Der Kreislauf beginnt von Neuem...**

## 4.2 Die Fachgebiete der Informatik

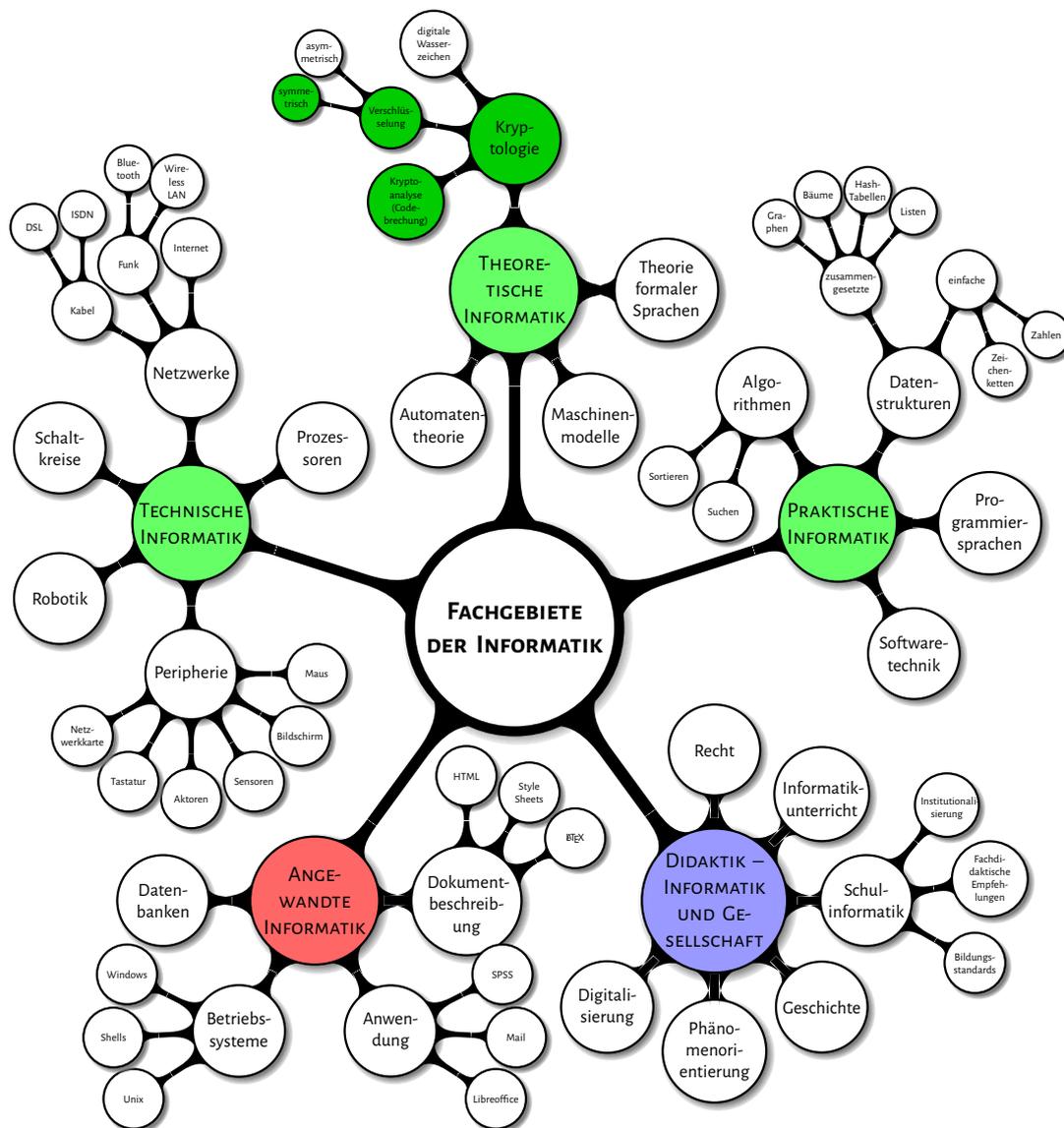


Abbildung 2: Die Fachgebiete der Informatik (vgl. Humbert, 2018) – Anpassung durch Fricke.

Die Informatik ist in sechs Fachgebiete unterteilt. Die *Technische Informatik*, die *Theoretische Informatik* und die *Praktische Informatik* werden zur sogenannten *Kerninformatik* (in Abb. 2 hellgrün) zusammengefasst. Dem gegenüber stehen die *Angewandte Informatik*, sowie *Informatik und Gesellschaft* und die *Didaktik der Informatik*, letztere sind in Abb. 2 in einen Oberpunkt zusammengefasst. Die Zuordnungen der Inhalte des vorliegenden Moduls sind *dunkelgrün* hervorgehoben. Abb. 2 ist als Übersicht zu verstehen. Sie lässt sich erweitern und ausdifferenzieren.

### 4.3 Zu erwerbende Kompetenzen im Rahmen des Moduls Kryptologie

Die Kompetenzen dieses Moduls wurden in Anlehnung an (Schüller, 2014, S. 27) formuliert. Mit den *Kompetenzen für informatische Bildung im Primarbereich* (Gesellschaft für Informatik e. V., 2019) werden die allgemeinen Informatikkompetenzen für den Primarbereich dokumentiert. Es handelt sich zum Einen um Kompetenzen im Rahmen einer Einführung in die Informatik, zum Anderen um Kompetenzen für das Modul Kryptologie.

Die Betrachtung dynamischer Aspekte, also Abläufe, in der Lebenswelt der Schülerinnen und Schüler sind Ausgangspunkt für den informatischen Problemaufschluss in der Grundschule.

#### Kompetenzen zur Informatik in der Lebenswirklichkeit

Die Schülerinnen und Schüler ...

- (IL-K1) beschreiben ihren persönlichen Umgang mit Informatiksystemen.
- (IL-K2) beschreiben ihren persönlichen Umgang mit Diensten des Internets.
- (IL-K3) beschreiben Analogien und Unterschiede zwischen der realen und der digitalen Welt.

#### Kompetenzen zur Kryptologie

Die Schülerinnen und Schüler ...

- (K-K1) charakterisieren den Begriff *Algorithmus*.
- (K-K2) erläutern gegebene Algorithmen.
- (K-K3) führen Algorithmen schrittweise aus.
- (K-K4) kennen ein Verfahren zur Transposition.
- (K-K5) analysieren den Algorithmus zur Verschlüsselung von Nachrichten (Daten).
- (K-K6) entwerfen einen Algorithmus zur Verschlüsselung von Nachrichten (Daten).
- (K-K7) »knacken« verschlüsselte Nachrichten mit Hilfe eines bekannten Verfahrens.
- (K-K8) erarbeiten eine Methode zur systematischen Entschlüsselung von verschlüsselten Nachrichten (Daten) und wenden diese an.

## 5 Grundlegende Sachinformation zur Kryptologie

Nach Borys gibt es zwei Möglichkeiten, den Zugriff auf Daten zu verhindern. Die erste Form ist das bloße *Verbergen* der Daten: Die Nachricht wird dadurch geschützt, dass sie nicht sofort sichtbar ist.

Die zweite Möglichkeit ist die Verschleierung oder *Verschlüsselung* der Daten. Durch Unkenntlichmachen der Daten ist der Zugriff darauf für Nichteingeweihte äußerst umständlich. Diese Methode nennt man *Kryptografie*. In dem vorliegenden Unterrichtsmodul werden die beiden symmetrische Verfahren zur Verschlüsselung, *Transposition* und *Substitution*, verwendet.

Ein asymmetrisches Verfahren, wie es etwa beim Verschlüsseln von E-Mails oder Online-Banking zum Einsatz kommt, wird in der abschließenden Unterrichtseinheit angerissen, jedoch fachlich nicht weiter vertieft. An dieser Stelle besteht jedoch Potential für eine Fortführung des Gelernten.

Die *Kryptoanalyse* widmet sich dem *Entschlüsseln* von Daten. Mit dem Entschlüsseln geht häufig die Intention einher, an die vor Zugriffen geschützten Daten zu gelangen. Dies geschieht meist durch Angriffe, die man als *Hacking* oder *Code-Breaking* bezeichnet. Die *Kryptologie* ist die Wissenschaft, die sich mit dem gesamten Themenfeld von der Verschlüsselung, über die Entschlüsselung bis hin zum Hacking und Code-Breaking beschäftigt (vgl. Borys, 2011, S.60f). *Codierung* kann hierbei als ein grundlegendes Verfahren zur Verarbeitung von Daten betrachtet werden.

### Überblick Verschlüsselung

Verschlüsselung und Codierung sind zentrale informatische Gegenstände, die in der heutigen, digitalen Kommunikation die zentrale Grundlage der Datenverarbeitung darstellen. Für Schülerinnen und Schüler kann diese zentrale Idee der Informatik ganz ohne Informatiksystem vermittelt werden.

Verschlüsselung bedeutet, dass ein *Schlüssel* verwendet wird, um die Zeichen oder Buchstaben der *Klartextbotschaft* zu repräsentieren. In diesem Fall muss der Schlüssel zwischen *Sender* und *Empfänger* bekannt sein. Durch Verschlüsselung können Botschaften sicherer geheim übermittelt werden. Jede *Geheimschrift* wird zunächst definiert durch ein bestimmtes Vorgehen, dem Verschlüsselungsalgorithmus. Dieser *Algorithmus* wird verknüpft mit einem *Schlüssel*, welcher nur dem Sender und dem Empfänger bekannt ist. Kombiniert ergibt sich ein System, welches Sender und Empfänger ermöglicht, geheim miteinander zu kommunizieren.

## 5.1 Steganographie

Im Rahmen des Moduls setzen sich die Schülerinnen und Schüler zunächst mit der Fragestellung auseinander, wie eine Nachricht verborgen werden kann. Sie beschäftigen sich dazu mit historischen Beispielen der »Steganographie«. Die Protagonisten Alice, Bob und ihre Gegenspielerin Eve begleiten die Schülerinnen und Schüler durch diese Lernaufgabe.

### Alice, Bob und Eve

Die Identifikationsfiguren Alice  und Bob  sind in der Kryptologie eine feste Institution zur Vermittlung von Szenarien: Daten werden von A (Alice) zu B (Bob) gesendet, der Angreifer Eve  (in Anlehnung an englisch: *evil*) versucht, die Nachricht abzugreifen.

Während Alice und Bob den Nachrichtenaustausch darstellen, stellt das Auftreten von Eve die Datensicherheit auf die Probe.

*Im Sinne einer kulturellen Vielfalt besteht die Möglichkeit, dass Sie die Darstellungen für Alice, Bob und Eve ersetzen, um mit deren äußerlichen Erscheinungen die kulturelle Vielfalt an Schulen aufzugreifen.*

Das Wort Steganographie setzt sich aus dem beiden griechischen Worten *steganos*: »bedecken« und *graphein*: »schreiben« zusammen (vgl. Singh, 2014, S. 20).

Die geheime Nachricht wird hierbei nicht codiert oder verschlüsselt, sondern es wird verborgen, dass die Nachricht überhaupt existiert (vgl. Singh, 2014, S. 20f). Es ist aus didaktischen Gründen ratsam, nicht vom *Verstecken* einer Nachricht zu sprechen, da die Schülerinnen und Schüler dadurch eventuell das Konzept entwickeln, Nachrichten würden in einem Objekt deponiert, bis der Adressat sie an sich nimmt. Beim *Verbergen* geht es vielmehr darum, die Nachricht vor Zugriff zu schützen, indem sie nicht mehr sichtbar ist. Daher besteht – und bestand auch historisch anfänglich – keine Notwendigkeit der Codierung oder Verschlüsselung. Eine Kombination aus Verbergen und Verschlüsseln erhöht jedoch zusätzlich die Sicherheit.

Im Unterrichtsverlauf entwickeln die Schülerinnen und Schüler zunächst eigenständig Ideen, wie man Nachrichten vor dem Zugriff anderer verbergen kann. Im Anschluss daran werden Sie mit historischen Varianten des Verbergens konfrontiert und überlegen, inwiefern Nachrichten verborgen wurden.

So etwa gab es Wachstafeln mit einer Nachricht auf dem Rahmen unter dem Wachs; Sklaven wurden die Köpfe rasiert und ihnen wurden Botschaften auf die Kopfhaut tätowiert; Eier wurden chemisch präpariert und Vieles mehr.

Hierbei entwickeln die Kinder ein Bewusstsein dafür, dass bereits in historischen Kontexten das Bedürfnis nach Geheimhaltung sensibler Daten bestand und dadurch dem Verbergen von Nachrichten eine hohe Bedeutung zukam.

### Verstecken vs. Verbergen

Sie können zur Vermittlung der Unterscheidung zwischen *Verstecken* und *Verbergen* im Unterricht die folgenden Formulierungen verwenden:

- ▶ »Ein Versteck ist ein Ort, an dem ich etwas ablege. Wenn ich einen Gegenstand verstecke, bringe ich ihn an diesen Ort.«
- ▶ »Verbergen bedeutet, dass ich so tue als ob eine Nachricht gar nicht da ist. Ich trage die Nachricht dort an meinem Körper, wo niemand nach ihr sucht.«

Auch unsere Wortspeicherkarten im Materialpaket (Vorschau auf S. KR 55) geben Ihnen Anregungen zur Vermittlung von Fachbegriffen in Ihrem Unterricht.

## 5.2 Codierung

Unter Codierung versteht man die Umwandlung einer Darstellungsform in eine andere. So können z. B. die Buchstaben einer Nachricht in Morse-Code umgewandelt werden. Die so codierte Nachricht ist nicht geheim, denn der Algorithmus für die Codierung, also wie die Nachricht codiert wird, ist allgemein bekannt.

Somit handelt es sich bei jeder Codierung um eine Konvention (vgl. Strecker, 2009).

Streng genommen ist auch das gesprochene Wort »Hund« bereits eine Codierung. Denn das mentale Abbild eines Hundes wird von uns mit einem sprachlichen Code, also dem verbal gesprochenen Wort »Hund« codiert.

Das gesprochene Wort »Hund« kann dann wiederum als Schriftbild codiert werden. Somit ist der Hund bereits doppelt codiert. Wenn das Wort »Hund« dann z. B. in einen Morse-Code übersetzt wird, ist der Hund sogar dreifach codiert.

Kinder lernen daher bereits während des Spracherwerbs zu codieren. Ab dem ersten Schuljahr lernen sie eine weitere Art der Codierung: indem sie die Schriftsprache erwerben, codieren sie Laute in Buchstaben bzw. Phoneeme in Grapheme.

Auch im Alltag kommen Kinder stetig mit den unterschiedlichsten Codierungen in Kontakt. So kennen sie möglicherweise im Rahmen von Spionage-Themen in Kinderzeitschriften bereits das Morse- Alphabet oder haben im Aufzug und anderen Bereichen des öffentlichen Lebens Blindenschrift wahrgenommen.

Aber auch in ihrer unmittelbaren Umgebung kommen sie mit Codierungen in Kontakt, ohne dies bewusst zu bemerken. Sie nehmen z. B. Verkehrsschilder wahr, ohne unmittelbar deren Bedeutung, also deren Codierung, zu kennen. Nachdem sie einzelne Schilder im Rahmen der Radfahrausbildung kennengelernt haben, kennen sie die Codierung und können sich verkehrskonform nach ihnen richten.

Auch Noten für ein Instrument sind eine Art von Codierung, mit der Kinder bereits im Grundschulalter in Kontakt kommen. Darüber hinaus gibt es in der kindlichen Lebenswirklichkeit noch Unmengen weiterer Codierungen, wie etwa QR-Codes oder Strichcodes, um nur zwei Beispiele zu nennen, die im Rahmen des Unterrichts thematisiert werden können.

In dem vorliegenden Modul machen die Schülerinnen und Schüler konkrete Erfahrungen mit Codierung am Beispiel der *Freimaurer-Chiffre*. Hierbei handelt es sich um eine Codierung, die den Zweck der Geheimhaltung hatte. Alternative Zugänge mit perspektiv- und fächerübergreifenden Bezügen, wie etwa der *Morse-Code* oder die *Blindenschrift*, bieten sich ebenfalls an.

### Nachricht oder Daten?

In der Unterrichtsplanung dieser Handreichung wird der Begriff *Daten* im Unterricht häufig durch das Wort *Nachricht* ersetzt. Die Schülerinnen und Schüler nutzen Codierungen und Verschlüsselungen an konkreten Beispielen – an Nachrichten, die sie verschicken wollen. Daher wird der Begriff *Nachricht* im Unterricht synonym verwendet.

### Freimaurercode

Der Geheimbund der Freimaurer versteht sich als ein weltweiter Zusammenschluss von Menschen, die mit ihrem Arbeitseinsatz den fünf Grundsätzen der Freimaurer (Freiheit, Gleichheit, Brüderlichkeit, Humanität und Toleranz) dienen.

Die Freimaurer sind in Logen organisiert und haben sich neben den genannten Grundsätzen darüber hinaus auch der Verschwiegenheit verschrieben. Angelegenheiten der einzelnen Logen sollen nicht nach außen getragen, wohl aber untereinander ausgetauscht werden.

Zu diesem Zwecke benutzten die ersten Freimaurer ein besonderes Chiffriersystem, also eine Geheimschrift. Bei der sogenannten *Freimaurer-Chiffre* handelt es sich um eine mono-alphabetische Substitution, wengleich die Buchstaben der Klarbotschaft nicht durch Buchstaben, sondern durch festgelegte Symbole ersetzt werden. Daher handelt es sich (auch) um eine Codierung, die aber immer wieder neu festgelegt werden kann.

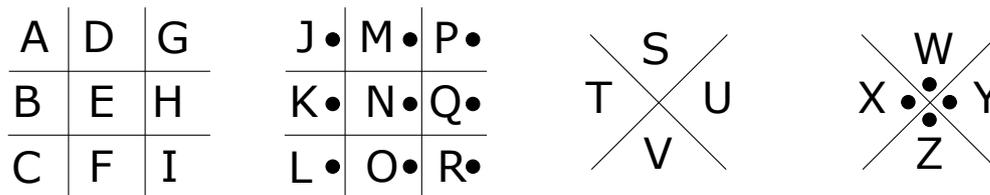


Abbildung 3: Beispiel für eine Freimaurer-Chiffre. Eigene Abbildung.

Die Grundlage für dieses Chiffrierverfahren beruht auf einem vereinbarten Code, welchen die Freimaurer an jedem Ort und zu jeder Zeit für die Dechiffrierung nutzen konnten. Die Grundlage dieser Chiffre bilden vier verschiedene Raster, die jeden Buchstaben des Alphabetes einer genauen Position zuordnen. Diese genaue Position ist durch die umgebenden Linien des Rasters und von eingefügten Punkten gekennzeichnet und damit eindeutig dem Buchstaben zuzuordnen.

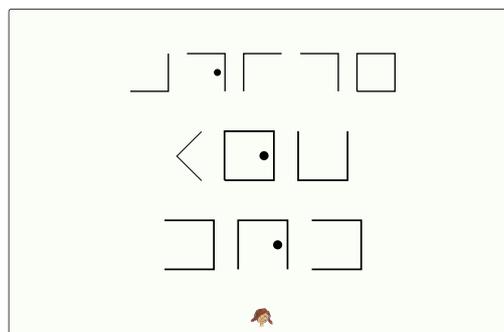


Abbildung 4: Beispiel für eine Codierung mit Freimaurer-Chiffre: »Alice und Bob«. Eigene Abbildung.

### 5.3 Transposition

Das Wort Transposition ist abgeleitet von dem lateinischen Wort *transponere*: »verschieben«.

Man spricht von einer *Transposition*, wenn bei der Verschlüsselung keine Zeichen weggelassen oder hinzugefügt, sondern die vorhandenen Zeichen nur verschoben (»transponiert«) werden (vgl. Singh, 2014, 29f).

Das bedeutet, eine Geheimbotschaft ist nichts weiter als ein Anagramm der Klartextbuchstaben der unverschlüsselten Nachricht. Dabei vergrößert sich die Anzahl der Möglichkeiten für die Geheimbotschaft, je mehr Zeichen die Klartextbotschaft enthält. Für eine Nachricht aus beispielsweise drei Buchstaben gibt es sechs verschiedene Möglichkeiten der zufälligen Transposition (siehe Infobox); bei vier Buchstaben sind es bereits 24 Möglichkeiten.

#### Beispiel einer Transposition: EVA

**Klartextbotschaft:** »EVA«

**Mögliche Anagramme als Geheimbotschaft:**

»EAV, VEA, VAE, AEV, AVE, EVA.«

Für Nachrichten, die aus noch mehr Buchstaben bestehen, entstehen so auch mehr Anagramme, so dass selbst computergestützte Verfahren zum Code-Knacken äußerst viel Zeit benötigen.

Daraus ergibt sich die Notwendigkeit, dass auch für die Transposition bestimmte Regeln (Algorithmen) bekannt sein müssen, damit der Empfänger die Nachricht entschlüsseln kann.

#### Skytale

Die Skytale gilt als erstes militärisch-kryptographisches Verfahren. Sie wurde bereits von den Spartanern im 5. Jahrhundert v. Chr. benutzt.

Die Skytale ist zunächst ein einfacher Holzstab. Zum Verschlüsseln wird ein schmaler Streifen Papier, Leder, Pergament oder Ähnliches um den Stab gewickelt. Die einzelnen Windungen des Streifen kommen nebeneinander zum Liegen (s. Abb. 5).



Abbildung 5: Skytale mit Lederriemen (Luringen, 2007) – Abbildung steht unter CC-BY-SA Lizenz.

Nun wird der Streifen auf der Skytale so beschriftet, dass die einzelnen Buchstaben der Klartextbotschaft auf den nebeneinander liegenden Abschnitten des Streifens stehen.

Wird nun dieser Streifen wieder abgerollt, entsteht daraus eine scheinbar willkürliche Buchstabenfolge – die Buchstaben verschieben sich in ihrer Reihenfolge und die Nachricht wird somit verschlüsselt. Das Auf- und Abwickeln auf einen Stab ist somit der Algorithmus. Nur mit einer Skytale mit dem gleichen Durchmesser kann die Geheimbotschaft durch erneutes Aufwickeln wieder in die Klartextbotschaft entschlüsselt werden.

#### Weitere Beispiele zur Transposition

Die weiteren Beispiele liefern Anregungen, um bei Bedarf oder Interesse den jeweiligen Unterrichtsgegenstand zu vertiefen.

**Beispiel für die Gartenzaun-Transposition**

**Klarbotschaft:** »In der Schule wird gelernt.«

**Modellierung zur Codierung:** »INDERSCHULEWIRDGELERNT«

**Gartenzaun-Transposition:**

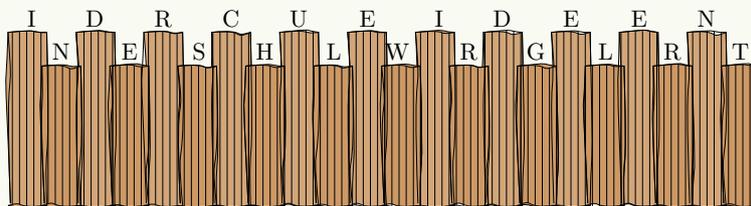


Abbildung 6: Gartenzaun. Eigene Abbildung.

**Geheimbotschaft:** »IDRCUEIDEENNESHLWRGLRT«

**Gartenzaun-Transposition**

Bei der Gartenzaun-Transposition werden die Buchstaben der Nachricht abwechselnd auf zwei Zeilen geschrieben. Als Geheimbotschaft wird nun zunächst die Buchstabenfolge der ersten Zeile aufgeschrieben, dann folgt die Buchstabenfolge der zweiten Zeile.

**Schablone**

Für die Verschlüsselung mit einer Schablone müssen sich Sender und Empfänger eine übereinstimmende Lochschablone herstellen. Grundlage dieser Lochschablone ist ein kariertes, quadratisches Feld mit z. B. 5 mal 5 Kästchen.

Dieses quadratische Feld muss zum Einen als Blanko-Exemplar für die Entschlüsselung der Nachricht zur Verfügung stehen. Zum anderen müssen aus diesem quadratischen Feld einzelne Felder ausgeschnitten werden, um so eine Lochschablone herzustellen, welche nur bestimmte Kästchen freigibt.

Mit dieser Lochschablone wird nun verschlüsselt, indem sie auf das Blanko-Feld gelegt wird und in die Löcher der Schablone die Buchstaben der Botschaft fortlaufend geschrieben werden. Die Schablone ist so gestaltet, dass bei dreimaligem Rotieren kein darunterliegendes Feld doppelt genutzt wird.

Wenn in allen Löchern ein Buchstabe auf dem Blanko-Feld steht, wird die Schablone um 90° gedreht und die nächsten Geheimtextbuchstaben werden in die dadurch freigelegten Kästchen eingetragen. So wird fortgefahren, bis alle vier Positionen der Schablonen verwendet worden sind.

	W		I			W	F	I		S	W	F	I		S	W	F	I	B
		R				F	R		E		F	R	U	E	E	F	R	U	E
T					T			N		T			N	M	T	N		N	M
	R					R			U		R	S		U	E	R	X	S	U
				E	N				E	N	I		E	E	N	I	R	E	E

Abbildung 7: Transposition mit Schablone (vgl. Müller, 2012) – Abbildung steht unter CC-BY-NC-SA Lizenz.

Wenn die Nachricht länger ist als die zur Verfügung stehenden Felder, wird ein weiteres Blanko-Quadrat benutzt. Ist sie kürzer, so werden die freien Felder einfach mit beliebigen Buchstaben gefüllt.

Bei ungerader Spalten- und Zeilenzahl darf das Kästchen in der Mitte nicht ausgeschnitten werden, da es sonst mehrfach belegt wäre. Dann bleibt in der Mitte ein Feld unbenutzt, da es durch keine Schablonenstellung belegt werden kann. In dieses Feld wird ein beliebiger Buchstabe eingetragen.

### 5.4 Substitution

Der römische Feldherr Julius Caesar (100 bis 44 v. Chr.) verschlüsselte seine geheimen Nachrichten, indem er jeden Buchstaben durch einen anderen ersetzte. Dabei wurde der Buchstabe immer durch den um eine bestimmte Anzahl von Stellen im Alphabet verschobenen Buchstaben ersetzt (vgl. Abb. 8). Diese Anzahl der Stellen heißt *Caesar-Schlüssel*, das Verfahren nennt sich *Caesar-Chiffre* bzw. *Caesar-Verschlüsselung*.

Klarbuchstabe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Geheimbuchstabe	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Abbildung 8: Caesar-Chiffre mit Verschiebung um drei Stellen. Eigene Abbildung.

Man spricht bei dieser Art der Verschlüsselung von einer *mono-alphabetischen Substitution*. Mono-alphabetisch bedeutet hierbei, dass nur ein einziges, festgelegtes Alphabet zum Einsatz kommt. Als Substitution wird dieses Verfahren bezeichnet, da die einzelnen Zeichen durch vorhandene Zeichen ausgetauscht bzw. ersetzt werden (lat.: »substituere«). Praktisch lässt sich das Verfahren, wie in Abb. 9 dargestellt, durch zwei sich überlagernde Scheiben realisieren.

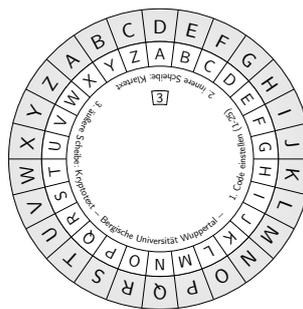


Abbildung 9: Caesar-Chiffre mit Scheiben (vgl. Müller, 2012) – Abbildung steht unter CC-BY-NC-SA Lizenz.

#### Weitere Beispiele zur Substitution

#### Caesar-Chiffre mittels Code-Wort

Mit dem Einfügen eines vereinbarten Codewortes entsteht ein Verschlüsselungssystem, welches von Unbeteiligten nur deutlich schwerer entschlüsselt werden kann. Zwischen Sender und Empfänger wird ein Codewort vereinbart und in der Buchstabentabelle den Geheimtextbuchstaben vorangestellt.

Die übrigen Buchstaben des Alphabetes, welche in diesem Wort nicht vorkommen, werden dann in der Reihenfolge des Alphabetes in die Tabelle geschrieben. Dabei können zwei Varianten auftreten.

### Erste Variante

Das Codewort besteht aus lauter unterschiedlichen Buchstaben, wie z. B. das Wort *Blumentopf*. Es wird der erste Buchstabe des Codewortes dem ersten Buchstaben des Alphabetes zugeordnet (vgl. Abb. 10). Der zweite Buchstabe des Codewortes wird dann dem zweiten Buchstaben des Alphabetes zugeordnet usw. Ist das Codewort zu Ende, werden die übrigen Buchstaben des Alphabetes in ihrer richtigen Reihenfolge eingetragen.

Klarbuchstabe	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
Geheimbuchstabe	B	L	U	M	E	N	T	O	P	F	A	C	D	G	H	I	J	K	Q	R	S	V	W	X	Y	Z	
Codewort	Blumentopf →																										

Abbildung 10: Caesar-Chiffre mit Codewort und anschließendem Auffüllen. Eigene Abbildung.

### Zweite Variante

Im Codewort tritt ein Buchstabe mehrfach auf (z. B. zwei »a« in *Apfelbaum*). In diesem Fall wird von jedem doppelten Buchstaben der erste eingetragen, der zweite jedoch entfällt. In diesem Fall wird aus *Apfelbaum* *Apfelbum*. Das weitere Vorgehen entspricht der ersten Variante.

## 5.5 Verschlüsselung im Alltag

Zum Abschluss der Unterrichtsreihe werden die Schülerinnen und Schüler wieder zum Ausgangspunkt zurück geführt. Sie kennen nun kryptologische Verfahren wie *Codierung* oder *Transposition* und haben durch *Eve* bereits Angriffsszenarien wie *Brute-Force* kennen gelernt. Abschließend werden sie dafür sensibilisiert, in welchem Alltagszusammenhang komplexe Verschlüsselung bei ihnen oder ihren Eltern auftaucht oder im Hintergrund abläuft und können ihr Wissen über Angriffe auf Verschlüsselung auch bei diesen Alltagserfahrungen einbringen. In dieser Einheit geht es weniger um das Verschlüsselungsverfahren, sondern vielmehr um das Angriffsszenario.

## 5.6 Glossar

Dieses Glossar ist im Rahmen der Lehrerschulungen mit den Korrespondenzschullehrkräften entstanden. Hinter dem Glossar steht die Absicht, Lehrkräften zur eigenen Vorbereitung und im Rahmen des Unterrichts fachsprachliche Anleitung zu den auftretenden Fachbegriffen zu geben.

Das vorliegende Glossar ist in Abgleich zu den Glossareinträgen der »Kompetenzen für informatische Bildung im Primarbereich« entstanden. Dort finden sich weitere fachlich vertiefende Glossareinträge.

⊕ Gesellschaft für Informatik e. V., Hrsg. *Kompetenzen für informatische Bildung im Primarbereich. Empfehlungen der Gesellschaft für Informatik e. V. Erarbeitet vom Arbeitskreis »Bildungsstandards Primarbereich«* – Beschluss des GI-Präsidiums vom 31. Januar 2019 – wird in gedruckter Form der LOG IN 39 (2019) Heft 191/192 beigelegt. 7. Feb. 2019. URL: <http://uni-w.de/1gm> (besucht am 09. 03. 2019)

**algorithmischer Grundbaustein** Bei der Beschreibung eines **Algorithmus** treten verschiedene Elemente auf. Diese Elemente können z. T. als Anweisungen bezeichnet werden. Eine Abfolge wird als *Sequenz* von Anweisungen dargestellt, die nacheinander abgearbeitet wird. Durch *Verzweigungen* und *Zyklen* (Wiederholung) wird es möglich, den Ablauf zu steuern, damit er flexibel gestaltet werden kann. Diese Elemente werden als Steueranweisungen oder als *Kontrollstrukturen* bezeichnet.

**Algorithmus** Ein Algorithmus ist eine eindeutige Handlungsabfolge.

Beispiel: Das Spiel »Mensch ärgere dich nicht!«:

1. Würfle *solange*, bis du eine 6 hast.
2. Gehe mit der ersten Figur auf »Start« (...)
3. *Wenn* du mit deiner Figur auf ein Feld kommst, auf dem bereits eine gegnerische Figur steht, *dann* gilt diese als geschlagen. (...)

Andere Beispiele für Algorithmen sind: Wegbeschreibung, Programmierung, Choreografie, Musikstück, Bastelanleitung, etc.

Eigenschaften eines Algorithmus:

1. eindeutig.
2. endlich.
3. terminiert: Der Algorithmus hört auf und läuft nicht unendlich weiter.
4. gleiche Eingabe = gleiche Ausgabe: Der Algorithmus muss bei gleicher Eingabe immer gleich durchlaufen werden.
5. Die Anweisungen sind ausführbar.

Diese fünf Eigenschaften kennzeichnen alle Algorithmen .

**Brute-Force** Brute-Force bezeichnet eine Methode des Knackens einer Verschlüsselung, die auf »roher Gewalt« (engl.: brute force) basiert. Durch vollständiges Ausprobieren aller Möglichkeiten erhält man irgendwann denjenigen **Schlüssel**, der den Klartext liefert.

**Caesar-Scheibe** Die Caesar-Scheibe stellt ein Hilfsmittel zur Durchführung einer **Verschlüsselung** mit dem **Substitutionsverfahren** dar. Hierbei wird auf der Scheibe eine Zahl als Schlüssel eingestellt. Daraus resultiert eine Zuordnung der Buchstaben in **Klarschrift** und **Geheimschrift**. Somit kann die Klarschrift *verschlüsselt* werden und später die *Geheimschrift* entschlüsselt werden.

**Code** Unter Code verstehen wir im informatischen Sinne die Konvention, wie z. B. allgemeine Schriftzeichen durch andere Zeichen ersetzt werden. So kann beim Morsecode ein Buchstabe durch eine Aneinanderreihung der Zustände *lang* und *kurz* dargestellt werden, um dann z. B. per Lichtzeichen übermittelt werden zu können. Die Elemente eines Codes werden in Form einer Code-Tabelle strukturiert dargestellt. Eine *Code-Tabelle* ist per Definition öffentlich und allgemein zugänglich. Daher ist ein Code nie eine Verschlüsselung.

**Code-Breaking** Unter Code-Breaking versteht man das Nachvollziehen eines Verschlüsselungs-**Algorithmus** gefolgt von dem Aushebeln eines Sicherheitsmechanismus, wie etwa einer **Verschlüsselung**. Erfolgreiches Code-Breaking ermöglicht somit den Zugriff auf verschlüsselte Daten.

**codieren** Codieren findet statt, wenn mittels einer *Code-Tabelle* eine Nachricht aus einem lesbaren Text **Klartext** in eine codierte Form umgewandelt wird. So kann z. B. eine Nachricht in Blindenschrift dargestellt werden oder per Morse-Code übertragen werden.

Als *decodieren* bezeichnen wir den umgekehrten Vorgang: eine codierte Nachricht wird mittels Code-Tabelle in einen lesbaren Text umgewandelt. Die Blindenschrift oder der Morse-Code werden dann wieder durch Schriftzeichen dargestellt. Ist der Code *nur einer Gruppe von Menschen* bekannt, so handelt es sich um eine **Verschlüsselung** – in der Alltagssprache werden diese beiden Begriffe oft nicht getrennt.

**Codierung** Darstellungsvereinbarung – z. B. stellt ein Verkehrsschild die Codierung eines Hinweises für die Verkehrsteilnehmenden bereit. Um eine Codierung nutzen zu können, muss man die Vereinbarung kennen, die beschreibt, was die jeweilige Codierung bedeutet. In der Schriftsprache werden zwei Schreibweisen, nämlich »Codierung« und »Kodierung« verwendet. Für dieses Dokument wurde entschieden, die Schreibweise »Codierung« zu verwenden.

**Coding** *Coding* beschreibt im englischen Sprachraum die Tätigkeit des Programmierens, also ein Programm zu schreiben. Es hat im engeren Sinne nicht mit der deutschen Verwendung des Wortes ☹ **Code** zu tun. Der Zusammenhang ergibt sich durch die Nutzung des Begriffs Quell-Code als Bezeichnung für den Programmtext.

**decodieren** Siehe ☹ **codieren**.

**geheim** Als *geheim* bezeichnen wir grundsätzlich jene Form von Daten, die nicht öffentlich sind. Gerade der ☹ **Schlüssel** einer Verschlüsselung sollte unbedingt geheim gehalten werden, damit der Geheimtext nicht durch Unbefugte entschlüsselt werden kann.

**Geheimtext** Ein Geheimtext ist eine Nachricht, die mittels ☹ **Verschlüsselung** »unlesbar« gemacht wurde. Mit Hilfe eines *Verschlüsselungsverfahrens*, bei dem der ☹ **Schlüssel** und der ☹ **Algorithmus** bekannt sind, kann der Geheimtext durch die Entschlüsselung wieder in einen ☹ **Klartext** umgewandelt werden.

**hacken** Hacking (engl.: eindringen) oder *hacken* bedeutet das Aushebeln von Sicherheitsmechanismen oder sich unberechtigt Zugang zu einem ☹ **Informatiksystem** zu verschaffen.

**Informatik** *Informatik* ist die Wissenschaft, die sich mit Fragen rund um die automatische Verarbeitung von *Daten* beschäftigt. Die Bezeichnung ist aus den Begriffen *Information* und *Automatik* zusammengesetzt.

**Informatiksystem** Ein Informatiksystem ist die technische Umsetzung für Problemlösungen der ☹ **Informatik**. Ein Informatiksystem besteht üblicherweise aus drei Komponenten mit verschiedenen Aufgaben: Hardware, Software und Netzverbindungen. Beispiele für Informatiksysteme sind: Computer, Tablet-Computer, Smartphones, aber auch Drucker, Router, SmartTV, Geld-/Fahrkartenautomaten, u. v. m. Alle Informatiksysteme sind Automaten. Vorsicht: Die fachsprachliche Verwendung des Begriffs *Automat* in der Informatik ist die verkürzte Form für Automatenmodell.

**Klartext** Ein Klartext ist eine Nachricht, die weder ☹ **codiert** noch ☹ **verschlüsselt** ist. Alle Schriftzeichen haben ihre konventionelle Bedeutung.

**knacken** Als Knacken (ugs.) versteht man das Aushebeln eines Sicherheitsmechanismus, wie etwa einer Verschlüsselung. Während sich das ☹ **Hacken** auf Systeme bezieht, betrifft das Knacken alle Arten von Sicherheitsmechanismen (»Aushebeln«).

**Kryptologie** Der Begriff Kryptologie fasst alle Betätigungsfelder zusammen, die mit dem ☹ **Verschlüsseln** und *Entschlüsseln* sowie der Analyse von Verschlüsselungen (*Kryptographie*) und den Angriff auf Verschlüsselungen (*Kryptoanalyse*) zusammenhängen.

**Nachricht** Als Nachricht bezeichnen wir im Kontext von Unterricht die zu übermittelnden **Daten**. Da die Schülerinnen und Schüler meist Nachrichten schreiben, erscheint diese Verallgemeinerung sinnvoll. Hinweis: Als Fachbegriff wird **Nachricht** in anderen Informatikzusammenhängen anders belegt. Die Formulierung »Botschaft« wäre auch möglich.

**öffentlich** Ein Verfahren zur Verschlüsselung ist immer öffentlich – was nicht öffentlich ist, ist der **⊕ Schlüssel**. Nur wer das Verfahren *und* den Schlüssel kennt, kann einen Geheimtext entschlüsseln. Bei einer Codierung ist die Code-Tabelle öffentlich – jeder, der die Code-Tabelle kennt, kann die Nachricht decodieren und als Text lesen.

**Schlüssel** Eine Verschlüsselung besteht immer aus dem Verfahren des **⊕ Verschlüsselns** und dem Schlüssel. Der Schlüssel ist der Teil einer Verschlüsselung, der Unbefugten nicht bekannt ist. Bei der *Caesar-Scheibe* ist der Schlüssel die Zahl, bei einer Skytale ist es der Durchmesser des Stabes, etc..

**Schlüsseltausch** Soll eine **⊕ Nachricht** vor dem Ausspähen durch Dritte geschützt werden, so empfiehlt sich eine **⊕ Verschlüsselung**. Diese bringt das Problem mit sich, dass diejenigen, die die Nachricht empfangen und entschlüsseln sollen, den **⊕ Schlüssel** kennen müssen. Daher ist es nötig, Verfahren zum sicheren Schlüsseltausch über unsichere Kanäle zu entwickeln. Der Schlüsseltausch kommt eher in *asymmetrischen Verschlüsselungsverfahren* zum Tragen, bei *symmetrischen Verschlüsselungen* sprechen wir eher von einer *Vereinbarungen* auf einen gemeinsamen Schlüssel.

**Skytale** Unter der Skytale versteht man einen Holzstab mit runder oder kantiger Form, auf den beschriebene Streifen aus Papier oder Leder schräg aufgewickelt werden können. Die Skytale ist ein historisches Mittel zur **⊕ Verschlüsselung** von Nachrichten. Das Verschlüsselungsverfahren der Skytale ist die Transposition, der Schlüssel ist die Dicke des Holzstabs.

**Steganographie** Unter Steganographie versteht man die Kunst des Verbergens. Hierbei ist die Nachricht im **⊕ Klartext** vorhanden, ihre Existenz wird aber verschleiert. So kann eine Nachricht auf die Kopfhaut tätowiert sein und sie wird nicht entdeckt, da sie mit Haaren überwachsen ist.

**Substitution** Die Substitution ist ein **⊕ Verschlüsselungsverfahren**. Hierbei werden die bedeutungstragenden Einheiten einer Nachricht (Buchstaben) nach einem bestimmten Muster (**⊕ Algorithmus** und **⊕ Schlüssel**) durch andere Einheiten ausgetauscht. Die Einheiten bleiben somit an ihrem Platz, ihr Sinn ist aber entstellt. Nur wer das Verfahren und den Schlüssel kennt, kann die Bedeutung der Buchstaben wieder herstellen. Substitution bedeutet *Ersetzen*.

**Transposition** Die Transposition ist ein **⊕ Verschlüsselungsverfahren**. Hierbei werden die bedeutungstragenden Einheiten einer Nachricht (Buchstaben) in ihrer Reihenfolge verschoben, die Buchstaben behalten aber ihre Bedeutung. Dadurch wird die Nachricht unlesbar. Transposition bedeutet *Verschieben*.

**verbergen** Beim Verbergen wird eine vorhandene Nachricht vor dem Zugriff Unbefugter geschützt, indem ihre Existenz verschleiert wird. So kann eine verborgene Nachricht an einer Stelle sein, an dem man nie suchen würde, z. B. in der Innenseite eines Gürtels.

**verschlüsseln** Verschlüsseln ist die Tätigkeit, bei der eine Nachricht vom Klartext mittels eines **⊕ Verschlüsselungsverfahrens** und eines **⊕ Schlüssels** in einen Geheimtext umgewandelt wird.

**Verschlüsselung** Unter einer Verschlüsselung verstehen wir die Anwendung eines Verfahrens zum Schutz des Zugriffs Dritter auf Daten. Hierbei wird ein *Verschlüsselungsverfahren* umgesetzt. Dieses Verfahren ist ein **⊕ Algorithmus** zum Verschlüsseln. Es wird außerdem ein **⊕ Schlüssel** benötigt, um das Verfahren korrekt durchzuführen.

**verstecken** Beim Verstecken, in Abgrenzung zum **⊕ Verbergen**, wird eine Nachricht an einem geheimen Ort plaziert. Diese semantische Abgrenzung ist sehr fein und wird von Kindern meist nicht durchdrungen. Eine redundante Nutzung der Begriffe ist daher möglich.

## 6 Didaktische Hinweise

In diesem Kapitel möchten wir explizite didaktische Hinweise geben, deren Berücksichtigung sich zur Durchführung des vorliegenden Moduls als sinnvoll herausgestellt haben. Auch sind in diesen Abschnitt die vielzähligen Rückmeldungen und Anregungen unserer Korrespondenzschullehrkräfte bei der Erprobung des Moduls eingeflossen.

### 6.1 Gesprächsführung

Bei dem dargestellten Unterricht nimmt der kommunikative Austausch und Abgleich des Erfahrenen eine wichtige Rolle ein. Der Lehrkraft kommt dabei die Funktion eines moderierenden Lernbegleiters zu. Als angemessenes Mittel zur Gesprächsführung empfiehlt sich daher das *sokratische Gespräch*. Über eine fragend-entwickelnde Haltung gegenüber den Kindern werden sie von ihrem gegenwärtigen fachlichen Konzept abgeholt und mit den logischen Fehlschlüssen ihrer Denkmuster sanft konfrontiert, um so zu der beabsichtigten Erkenntnis geführt zu werden. Weiterführende Erläuterungen zum sokratischen Gespräch finden sich unter anderem bei Albers (Albers, 2013).

### 6.2 Sprachsensibel im Fachunterricht

Sprachsensibilität ist ein ebenso wichtiger Aspekt für den Fachunterricht. Neben der Vermittlung von fachlichen informatischen Inhalten und fachdidaktischen Methoden der Informatikdidaktik stellt sich die Frage des adressatengerechten Einsatzes von Fachsprache.

Wir sind überzeugt, dass sprachsensibler Unterricht im Lexikalischen beginnt. Hilfestellungen hierfür finden Sie in unserem Glossar (S. KR 15) und im schülerorientierten Wortspeicher (S. KR 55).

Auch Sprachmuster, Formulierungen und Satzteile und Sprachbausteine sollten immer wieder mündlich und schriftlich zur Verfügung gestellt, wiederholt und eintrainiert werden. Die Lehrkraft dient hierbei als Sprachvorbild und ermöglicht den Schülerinnen und Schülern ein Sprachbad.

Weiterführende Anregungen für sprachsensiblen Fachunterricht finden sich beispielsweise bei Yurdakal Cakir-Dikkaya (Yurdakul Cakir-Dikkaya u. a., 2017).

### 6.3 Allgemeine Hinweise zu Möglichkeiten der Differenzierung

Das vorliegende Modul versteht sich als *Planungshilfe* für Ihren Unterricht. Dies bedeutet, es steht Ihnen frei, qualitativ und quantitativ zu differenzieren, um die Lerninhalte und den Verlauf des Unterrichts den Bedürfnissen Ihrer Lerngruppe anzupassen. Nachfolgend skizzieren wir Ihnen daher mögliche Vorgehensweisen zur Differenzierung.

- Zerteilen Sie Arbeitsblätter, um Phasen dynamischer zu gestalten.
- Lassen Sie Teile von Aufgaben weg.
- Ergänzen Sie Aufgaben.
- Ändern Sie Arbeitsaufträge ab.
- Bieten Sie Ihren Schülerinnen und Schülern weitere Beispiele aus dem Lehrerteil dieser Handreichung an, um Inhalte zu vertiefen (z. B. für Transposition: Gartenzaun, Schablone)
- Reichern Sie einzelne Phasen durch zusätzliche Materialien, z. B. Lesetexte, Sachbücher, Internetseiten oder Filme, an. In den weiterführenden Materialien (S. KR 56) finden Sie hierzu Anregungen.

## 7 Angebote zur Unterrichtsgestaltung – Modul Kryptologie

In diesem Kapitel finden Sie konkrete Planungen für die unterrichtliche Umsetzung.

Jede Unterrichtseinheit beginnt mit dem Ausweis der **Ziele und Kompetenzen** und einer Darstellung des **zeitlichen Umfangs**. In einigen Fällen sind **Lernvoraussetzungen auf der methodischen Ebene** angegeben.

Darauf folgt die **empfohlene Vorgehensweise**. Eine Abweichung von der vorgeschlagenen Umsetzung und die Anpassung an die Bedürfnisse der Lernsituation ist jederzeit möglich.

Zu guter Letzt gibt es in einer *separaten Materialsammlung* einen **Überblick** über die **Kopiervorlagen (KV)** und **Demonstrationsmaterialien** zur Unterrichtseinheit. Die Farbgebung gibt Aufschluss über die Art des Dokuments.

### Farbgebung der Materialien

#### Hinweise

Darstellung von **Hinweisen** als Rahmen für ein Dokument.

Hinweise sind **dunkelgrün**.

#### Arbeitsblätter und Materialien zur Auslage sowie Aufgaben

Darstellung der **Materialien und Aufgaben** als Rahmen für ein Dokument.

Arbeitsblätter und Materialien zur Auslage sowie Aufgaben mit methodischem Schwerpunkt sind **grün**.

#### Briefe der Protagonisten an die Klasse

Darstellung der **Briefe der Protagonisten an die Klasse** als Rahmen für ein Dokument.

Briefe der Protagonisten an die Klasse sind **hellgrün**.

In einzelnen Fällen muss bzw. kann **zusätzliches Material** miteinbezogen werden, dies wird entsprechend angegeben.

## Übersicht über die Unterrichtseinheiten

Einheit	Thema	Kurzbeschreibung	Lernziele	Zeit	Seite
Einstieg	<i>Informatik in der Lebenswirklichkeit</i> <b>Aktivierung</b>	Die Schülerinnen und Schüler erleben am Lernort Supermarkt, welche informatischen Phänomene im Alltag aufwarten und dokumentieren ihre Assoziationen	Informatikphänomene wahrnehmen; Vorwissen aktivieren und dokumentieren.	1 Einzelstunde	S. KR 24
1	<i>Steganographie</i> <b>Verbergen von Nachrichten</b>	Die Schülerinnen und Schüler lernen Alice, Bob und Eve kennen. Sie werden mit der Problemstellung konfrontiert, eine Nachricht zu verbergen.	Erproben und Beurteilen des Verbergens von Nachrichten; Wortschatzerweiterung.	1 Doppelstunde	S. KR 27
2	<i>Codierung</i> <b>Der Freimaurercode</b>	Die Schülerinnen und Schüler arbeiten mit dem Freimaurercode. Sie (de)codieren Nachrichten und bewerten das Verfahren.	Anwenden, Beschreiben und Beurteilen von Algorithmen und deren Teilschritten; Wortschatzerweiterung.	1 Doppelstunde	S. KR 33
2a	<i>Codierung</i> <b>Codes sind überall</b>  <i>optional</i>	Die Schülerinnen und Schüler definieren den Begriff <i>Code</i> und finden anhand von vorbereitetem Material weitere Codes aus ihrer Alltagswelt. Sie erstellen ein Klassenplakat zu Codierungen.	Vertiefen und Sichern des Verständnisses von Codes und Codierung. Wahrnehmung von Codes im Alltag.	1 Einzelstunde	S. KR 37

Einheit	Thema	Kurzbeschreibung	Lernziele	Zeit	Seite
3	<i>Transposition</i> <b>Die Skytale</b>	Die Schülerinnen und Schüler lernen entdeckend die Skytale und ihre Funktionsweise kennen und nutzen sie, um Nachrichten zu ver- und entschlüsseln.	Kennenlernen und Erproben eines Verfahrens zur Transposition; Durchführen und Beschreiben von Algorithmen; Wortschatzerweiterung.	1 Doppelstunde	S. KR 41
4	<i>Substitution</i> <b>Caesar-- Verschlüsselung mit Code-Scheibe</b>	Die Schülerinnen und Schüler nutzen eine Code-Scheibe zur Ver-/ Entschlüsselung wie Caesar.	Kennenlernen eines Verfahrens zur Substitution am Beispiel Caesar-Codescheibe; Beschreiben und Ausführen des Algorithmus.	1 Doppelstunde	S. KR 47
5	<i>Informatik in der Lebenswirklichkeit</i> <b>Verschlüsselung im Alltag</b>	Die Schülerinnen und Schüler erkennen Verschlüsselung in Alltagssituationen wieder und übertragen ihnen bekannte Angriffsszenarien.	Theoretische Anwendung von Angriffsszenarien auf Verschlüsselungsphänomene im Alltag.	1 Einzelstunde	S. KR 51

## Einkaufsliste

Die folgenden Materialien benötigen Sie *zusätzlich* zu den bereitgestellten Arbeitsblättern und Kopiervorlagen.

### Für alle Elemente des Moduls Kryptologie ...

- Plakat zum Festhalten von Fachbegriffen als Wandzeitung

### **Einstieg: Informatik in der Lebenswirklichkeit – Im Supermarkt**

- kein Zusatzmaterial notwendig

### **01 – Steganographie**

- Geheimstifte. 2016. URL: <https://t1p.de/IaG-Geheimstifte> (besucht am 09. 03. 2019).
- alte Baumwollsocken
- Wachs, z. B. aus Teelichtern
- kleine Tafel, z. B. aus dem Dekorationshandel; alternativ Grillschale aus Alufolie
- Tintenkiller
- permanent- und non-permanent Marker
- OHP-Folie
- Knete
- Stoffstreifen

### **02 – Codierung – Der Freimaurercode**

- kein Zusatzmaterial notwendig

### **02a – Codierung – Codes sind überall (optional)**

- ein Bogen Tonpapier DIN A2

### **03 – Transposition – Skytale**

- Rundhölzer in zwei Durchmessern für 2 halbe Klassensätze
- weitere Röhren und Stäbe in unterschiedlichem Durchmesser und Formen

### **04 – Substitution – Caesar-Verschlüsselung**

- Tonpapier DIN A4 in rot (8 Blatt) und grün (4 Blatt)
- Laminierfolien DIN A4 12 Stück
- Musterbeutelklammern: 16 Stück

### **05 – Verschlüsselung im Alltag**

- kein Zusatzmaterial notwendig

# Informatik in der Lebenswirklichkeit

## Im Supermarkt

Einstieg

## Informatik in der Lebenswirklichkeit



## Ziele und Kompetenzen

Die Schülerinnen und Schüler...

-  entwickeln eine Grundvorstellung verborgener, informatischer Prozesse.
-  erkennen automatische Prozesse in ihrer Umwelt.
-  beschreiben mit eigenen Worten informatische Prozesse in ihrem Alltag.
-  beschreiben Prozesse, die mit Informatiksystemen gestaltet werden.

## Zeit

 1 Einzelstunde

## Lernvoraussetzungen

 Placemat

## Empfohlene Vorgehensweise

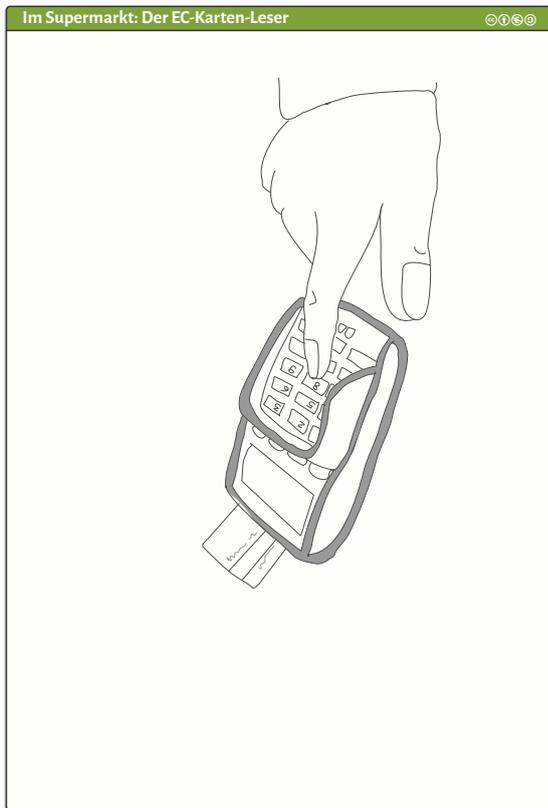
- ▶ Aktivieren Sie das Vorwissen der Schülerinnen und Schüler, indem Sie im Plenum Bilder aus dem Supermarkt zeigen (Scannerkasse, elektronische Waage, Barcodes) und dabei erläutern, dass viele informatische Prozesse im Hintergrund ablaufen. *Alternativ zum Supermarkt finden Sie Bildkarten zur Bushaltestelle.*
- ▶ Lassen Sie die Kinder in Gruppenarbeit ein Placemat mit weiteren Ideen zu informatischen Prozessen erstellen (elektronische Anzeigetafeln, Drehtüren, Rolltreppen, Ampelschaltungen, etc).
- ▶ Sammeln Sie die Ergebnisse der Kinder an der Tafel.
- ▶ Überlegen Sie im Plenum, wie diese informatischen Prozesse nach Auffassung der Schülerinnen und Schüler ablaufen.
- ▶ Stellen Sie heraus, dass für den Ablauf verborgener informatischer Prozesse Daten ausgetauscht und verarbeitet werden.
- ▶ Führen Sie im Anschluss einen kurzen Unterrichtsgang zu einer Ampel, einem Supermarkt, einer Bushaltestelle o. Ä. durch und untersuchen Sie mit den Schülerinnen und Schülern den ausgewählte Lerngegenstand auf informatische Prozesse.
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

## Zusätzliches Material

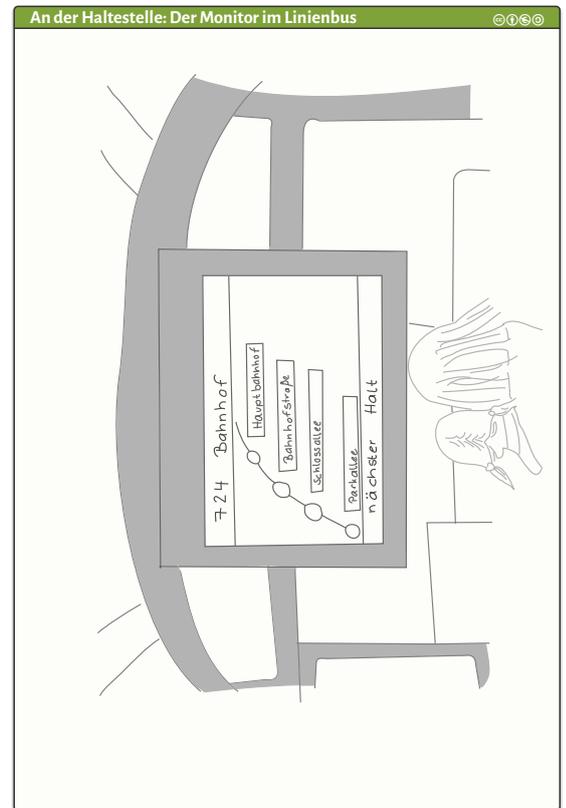
-  KV Bildkarten Supermarkt (oder Bushaltestelle)  
Hinweis: Drucken Sie **entweder** die KV Bildkarten zum Supermarkt **oder** zur Bushaltestelle je nach Unterrichtsgang aus.
-  KV Placemat
-  Wandzeitung

**Materialübersicht: Informatik in der Lebenswirklichkeit (Auszug)**

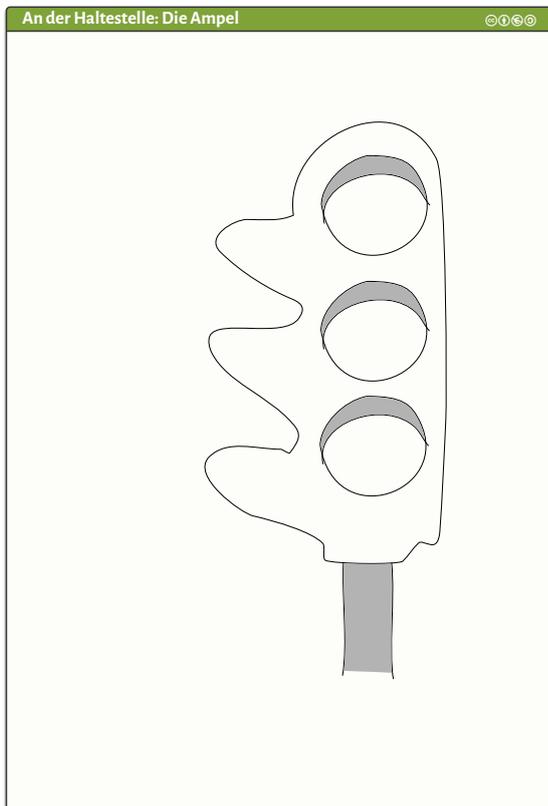
Für die Unterrichtsplanung ist die Entscheidung **Supermarkt oder Haltestelle** zu treffen.



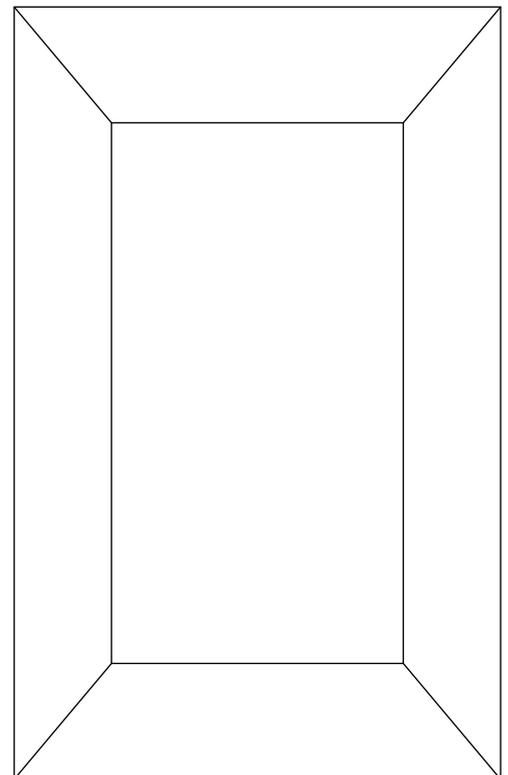
KV EC-Lesegerät



KV Bus-Display



KV Ampel



AB Placemat

# Steganographie

## Verstecken und Verbergen

Unterrichtseinheit 1

## Verstecken und Verbergen



## Ziele und Kompetenzen

Die Schülerinnen und Schüler wissen...

-  was das Wort *verbergen* im informatischen Kontext bedeutet.
-  dass Nachrichten verbergen werden können, um sie vor Zugriffen anderer zu schützen.
-  welche Möglichkeiten des Verbergens im Alltag anwendbar sind.
-  dass Nachrichten nur so lange verbergen sind, wie der Angreifer die Art des Verbergens nicht kennt.

## Zeit

 ca. 1 Doppelstunde

## Lernvoraussetzungen

 Think-Pair-Share oder Placemat

## Empfohlene Vorgehensweise

- ▶ Lesen Sie Brief 1 vor.
- ▶ Führen Sie die Protagonisten Alice und Bob und Eve ein. Hängen Sie dazu die Bilder der drei Charaktere an die Tafel und stellen Sie ein Angriffsszenario grafisch dar, so dass den Schülerinnen und Schülern klar wird, dass eine Nachricht von A nach B geschickt wird und dabei vor dem Zugriff von Eve geschützt werden soll.
- ▶ Führen Sie zur Problemstellung hin: *Wie kann man die Nachricht **verbergen**?* Klären Sie dazu unbedingt sprachlich den Unterschied zwischen **verstecken** und **verbergen**.
- ▶ Führen Sie mit den Schülerinnen und Schülern nun ein Think-Pair-Share oder alternative ein Placemat mit ihren Ideen durch.
- ▶ Lassen Sie die Schülerinnen und Schüler ihre Ideen im Plenum vorstellen. Lassen Sie sie dabei ggf. über die Qualität des Verbergens diskutieren.
- ▶ Teilen Sie den Schülerinnen und Schülern in Partnerarbeit die vorbereiteten Beispiele für Verbergen aus: *Geheimstifte, Tintenkiller, OHP-Marker, Knetkugel*. Sie können den Kindern auch eine fertige Kiste mit vorbereiteten, verborgenen Nachrichten geben und sie auffordern, diese zu entdecken.
- ▶ Stellen Sie im Anschluss historische Beispiele aus der Steganographie vor: *Sklavenkopf, Wachskugeln, Wachstafel*.
- ▶ Lesen Sie Brief 2 vor. Besprechen Sie mit den Schülerinnen und Schülern, dass durch das Finden der Nachricht der Inhalt direkt bekannt ist und wie man die Nachricht besser schützen kann.
- ▶ Regen Sie die Schülerinnen und Schüler für die nächste Stunde dazu an, sich Gedanken zu machen, wie eine Nachricht besser geschützt werden kann.
- ▶ Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.

## Verstecken und Verbergen



### Material- überblick

-  KV Brief 1
-  AB Think-Pair-Share
-  Erklärungshilfen zu den historischen Beispielen
-  KV Brief 2

### Zusätzliches Material

-  Stifte für Geheimschrift
-  Tintenkiller
-  OHP-Marker
-  Folie
-  Knete
-  Stoff, mit Nachricht beschriftet
-  Modell eines Sklavenkopfes
-  Wachskugel, mit verborgener Nachricht
-  ggf. Wachstafel
-  Wandzeitung

## Hinweis zur Materialerstellung: Steganographie



Nachfolgend wird erläutert, wie die Materialien zur Steganographie erstellt werden können.

### Material für die moderne Steganographie

**Geheimschrift** Für die Geheimschrift empfehlen wir Spezialstifte. Diese schreiben ähnlich wie ein Tintenkiller zunächst unsichtbar auf das Blatt. Mit dem integrierten Schwarzlicht im Stiftdeckel kann das Geschriebene schnell wieder sichtbar gemacht werden. Ein Link hierzu findet sich in der Einkaufsliste.

Weitere Möglichkeiten für Geheimschrift sind: Zitronensaft, Milch (oder Urin). Hierzu das Blatt mit einem Pinsel beschriften und anschließend über eine Flamme halten oder mit einem Bügeleisen erhitzen.

**Tintenkiller** Ein der Geheimschrift ähnliches Verfahren ist der Tintenkiller: Schreiben Sie mit dem Tintenkiller unsichtbar auf ein Blatt. Wenn danach mit Tinten darüber schraffiert wird, bleibt die Nachricht weiß auf dem Blatt zurück.

**OHP-Marker** Schreiben Sie auf einer OHP-Folie mit Permanent-Marker die Nachricht. Schraffieren Sie anschließend mit non-permanent-Marker darüber. Wenn Sie nun den non-permanent Marker mit Wasser abwischen, kommt die Nachricht wieder zum Vorschein.

**Knetkugel** Beschreiben Sie eine Stoffbahn mit Ihrer Nachricht. Wickeln Sie den Stoff fein auf und verbergen Sie ihn in einer Kugel aus Knete. Dieses Beispiel ist eine gute Überleitung zu den historischen Verfahren.

### Material für die historischen Beispiele der Steganographie

**Sklavenkopf** Die Kopiervorlage für den Sklavenkopf im Materialpaket laminieren und den Kopf ausschneiden. Von einem Baumwollsocken ca. 10cm inklusive Bündchen abschneiden (Ferse und Fußteil werden nicht benötigt). Die Nachricht mit Folienstift auf den oberen Teil des Kopfes schreiben. Den Sockenteil mit dem Bündchen über den laminierten Kopf ziehen, überstehenden Sockenrand als Haare in Streifen schneiden.

**Wachstafel** In Dekorationsgeschäften finden sich gelegentlich handliche Kreidetafeln, die präpariert werden können.

In jedem Fall wird die Nachricht auf die Innenseite der Tafel geschrieben, danach wird heißes Wachs darüber gegossen. Anschließend kann man zum Wiedersichtbarmachen der Nachricht mit einem Ceranfeld-Kratzer das Wachs abtragen.

Im Unterricht sollten Sie auf die ursprüngliche Verwendung von Wachstafeln eingehen, da dieses Schreibgerät nicht mehr in der Lebenswirklichkeit der Kinder vorkommt.

**Wachskugel** Für die Wachskugeln schreiben Sie auf ein Stück Stoff, oder historisch korrekt Seide, Ihre Nachricht. Danach wird der Stoff sauberlich zusammengerollt, so dass die innenliegende Fläche vor dem Wachs geschützt ist. Die so entstandene Kugel wird dann in flüssiges Wachs getaucht. Eventuell bietet sich aber auch eine Umhüllung, wie etwa eine Überraschungs-Ei-Dose ein gutes Modell. Zum Lesen der Nachricht kann man das Wachs der Kugel erneut erwärmen oder den Stoff auch kalt vorsichtig auseinander falten.

# Materialübersicht: Steganographie I

**Brief 1**



Hallo Kinder,  
ich bin Bob!

Mit mir könnt ihr immer wieder spannende und lustige Geschichten erleben.



Ich habe auch eine Freundin in der Schule. Sie heißt Alice. Wir schreiben uns gerne mal Briefe.



Ein anderes Mädchen in unserer Klasse ist Eve. Sie möchte immer unsere Briefe mitlesen. Wir haben überlegt, dass wir die Briefe irgendwie vor ihr verbergen müssen.

Habt ihr eine Idee, wie das gehen könnte?  
Wie kann man Nachrichten vor anderen Menschen verbergen?

Danke für eure Hilfe!  
Euer Bob

KV Brief1

**Brief 2**



Hallo Kinder,  
ich bin's, Alice.

Wow, ihr habt ja richtig gute Ideen zum Verbergen gehabt.

Und die Beispiele, die wir euch mitgebracht haben, konntet ihr ja auch supergut lösen! Spitzenmäßig!

Aber wisst ihr was?

Eve ist genauso schlau wie ihr!

Sie hat alle verborgenen Nachrichten gefunden und gelesen! Und dabei haben wir uns soviel Mühe gegeben, alles vor ihr geheim zu halten.

Wie können wir unser Nachrichten besser verbergen?  
Vielleicht habt ihr ja bis zur nächsten Stunde noch ein paar gute Ideen?

Viele Grüße,  
Eure Alice

KV Brief2

**Ich – Du – Wir – Gespräch**  
Methode Think-Pair-Share

**Aufgabe**



Wie kann man eine Nachricht verbergen?

**1. Schritt**  
Arbeite alleine.  
Schreibe deine Ideen auf.

**2. Schritt**  
Tausche dich mit deinem Sitznachbarn aus. Einigt euch auf **eine** gute Idee.  
Schreibt eure gemeinsame Idee auf.

**3. Schritt**  
Stellt eure Idee vor der Klasse vor.  
Welche andere guten Ideen hatten deine Klassenkameraden?  
Schreibe sie auch auf.

KV Think Pair Share

**Historische Beispiele der Steganographie**

**Der Sklavenkopf**

Schon damals bei den alten Griechen vor ungefähr 4000 Jahren war Geheimhaltung ein wichtiges Thema. Die Armee der Spartaner war auf dem Weg ein Königreich in Griechenland zu erobern! Doch es gab einen Spion. Er wusste von dem Angriff und schickte seinen Boten los. Der Bote durfte nicht enttarnt werden. Seine geheime Nachricht wurde gut verborgen. Er durchquerte viele Kontrollen, aber niemand fand die Nachricht.

Als er endlich an seinem Ziel angekommen war, wusste der König direkt, was zu tun war. Er las die Nachricht und bereitete sich auf den Angriff vor. Nur so konnte er sein Königreich schützen und hunderte Leben retten.

Doch wie wusste er, wo und wonach er suchen musste?  
Wo war die Nachricht verborgen?



Die Nachricht des Boten war sehr gut verborgen. Der Spion wählte einen Menschen mit vollem Haar. Er rasierte ihm den Kopf und ließ die Nachricht auf seine Kopfhaut tätowieren. Erst als die Haare wieder nachgewachsen waren, schickte er seinen Boten zum König, um diesen zu warnen.

Der König hatte damals eigenhändig den Spion ausgewählt und so wusste er, wo er nach der Nachricht suchen musste. Die Soldaten an den vielen Kontrollpunkten durchsuchten zwar alle Gegenstände, aber auf die Idee, jemandem den Kopf zu rasieren, waren sie bisher noch nicht gekommen.

Zum Glück dauerten die Kriegsvorbereitungen so lange, dass der Bote genug Zeit hatte, bis seine Haare nachgewachsen waren.

So eine Geheimbotschaft ist nichts für Ungeduldige!

Historische Beispiele zum Verbergen

# Materialübersicht: Steganographie II

**Historische Beispielen der Steganographie**

**Die Wachskugel**

Im alten China nahm man Seidenstreifen. Darauf wurde die Nachricht geschrieben. Anschließend rollte man die Seide und tauchte die so entstandene Kugel in flüssiges Wachs.

Nachdem die Wachskugel kalt geworden war, musste der Bote sie schlucken. Später ging er auf die Toilette und schied sie wieder aus.

Das Wachs wurde heiß gemacht und wenn es flüssig genug war, konnte man die Seidenkugel wieder ausrollen und die Nachricht lesen.

Na dann guten Appetit!

---

**Die Wachstafel**



Im Alten Rom vor etwa 2000 Jahren schrieb man nicht in Schulhefte oder auf dem Handy sondern auf Wachstafeln.

Dazu wurde ein Rahmen aus Holz geklebt, in den flüssiges Wachs gegossen wurde. Mit einem *Griffel* wurde wie mit einem Stift geschrieben, nur mit dem Unterschied, dass beim Schreiben Wachs abgekratzt wurde.

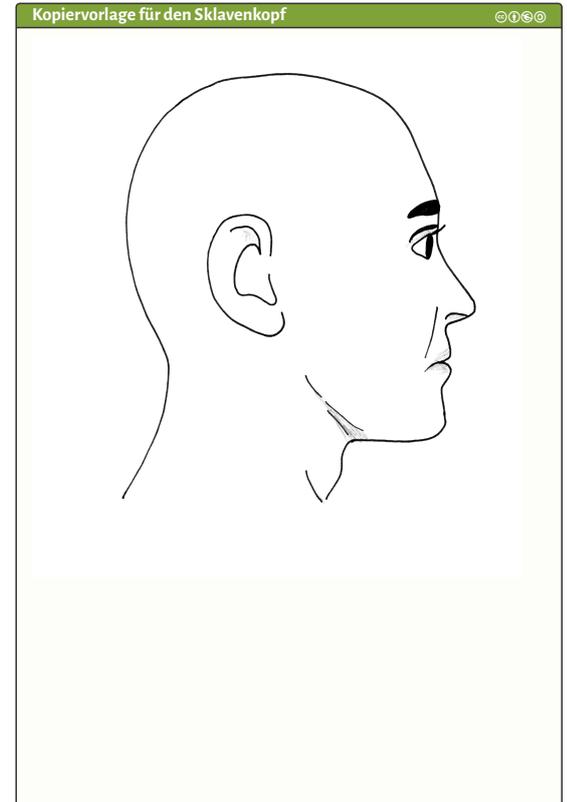
Und wo ist jetzt die *geheime* Nachricht versteckt?

Bevor man das Wachs in den Rahmen gießt, kann man auf die Innenseite des Holzrahmens eine Nachricht in das Holz schnitzen.

Die kann man erst wieder lesen, wenn man das ganze Wachs flüssig macht und wieder ausgießt.

Und wer würde sowas schon machen! Clever, oder?

Historische Beispiele zum Verbergen



KV Sklavenkopf

# Codierung

## Der Freimaurercode

Unterrichtseinheit 2

## Der Freimaurercode


**Ziele und  
Kompetenzen**

Die Schülerinnen und Schüler...

-  erläutern gegebene Algorithmen, indem sie beschreiben, wie sie Codierungstabellen nutzen.
-  führen Algorithmen schrittweise aus, indem sie Codierungstabellen einsetzen, um Nachrichten zu codieren und decodieren.
-  kennen die Fachbegriffe *Code*, *codieren* und *decodieren* und können ihnen einen Inhalt zuweisen.
-  nutzen steganographische Ansätze zum Verbergen codierter Nachrichten.

**Zeit**

 1 Doppelstunde

**Lern-  
voraussetzungen**

 Meeting-Points

**Empfohlene  
Vorgehens-  
weise**

- ▶ Aktivieren Sie das Wissen aus der letzten Stunde. Lassen Sie die Schülerinnen und Schüler berichten, welche weiteren Möglichkeiten zum Verbergen von Nachrichten sie sich ausgedacht haben.
- ▶ Lesen Sie den Schülerinnen und Schülern Brief 3 vor.
- ▶ Zeigen Sie den Schülerinnen und Schülern den codierten Text. Lassen Sie die Schülerinnen und Schüler diskutieren, wie man ihn lesen, also decodieren, kann. Helfen Sie den Kindern ggf. bei der Erkenntnis, dass es einer Codetabelle bedarf.
- ▶ Lassen Sie die Schülerinnen und Schüler in Partnerarbeit mit Hilfe der Code-Tabelle die Nachricht decodieren. Hängen Sie ggf. eine Kontrolllösung ins Tafelinnere.
- ▶ Lassen Sie die Schülerinnen und Schüler im nächsten Schritt in Partnerarbeit selbst Nachrichten codieren. Differenzieren Sie ggf. über den Umfang des Wortmaterials (ein Name, ein Wort, ein kurzer Satz, etc.). Wiederholen sie ggf. die Klassenregeln und Umgangsformen zur Vermeidung unhöflicher Wörter.
- ▶ Richten Sie Meeting-Points zum Austausch der codierten Texte ein und lassen Sie die Schülerinnen und Schüler gegenseitig ihre Nachrichten decodieren.
- ▶ Thematisieren Sie mit den Schülerinnen und Schülern die Vorgehensweise beim Einsatz von Code-Tabellen und lassen Sie die Schülerinnen und Schüler verbalisieren, wie sie vorgegangen sind.

## Der Freimaurercode

**Empfohlene Vorgehensweise**

- ▶ Lassen Sie die Schülerinnen und Schüler bewerten, wie sicher die Nachrichten nun sind. Bringen Sie dazu wieder Eve als Angreiferin ins Spiel. Lassen Sie die Schülerinnen und Schüler überlegen, was Eve tun müsste, um die Nachrichten lesen zu können.
- ▶ Die Schülerinnen und Schüler werden darauf kommen, dass Eve die Code-Tabelle benötigt, um alle Geheimnachrichten zu lesen.
- ▶ Lassen Sie die Schülerinnen und Schüler zum Abschluss der Stunde erneut überlegen, wie man die Nachrichten noch besser geheim halten könnte.
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

**Material-überblick**

-  KV Brief 3
-  KV Nachricht im Freimaurercode
-  KV Freimaurercode-Tabelle
-  AB Nachrichten codieren/decodieren

**Zusätzliches Material**

-  Wandzeitung

# Materialübersicht: Der Freimaurercode

**Brief 3**



Hallo Kinder,  
ich bin's wieder, eure Alice.

Wow, ihr hattet ja viele Ideen, wie man eine Nachricht vor dem Zugriff anderer schützen kann.  
Bob und ich haben auch noch einmal nachgedacht und ein bisschen recherchiert.

Dabei sind wir auf die Freimaurer gestoßen.  
Die Freimaurer waren eine Vereinigung von Handwerkern vor einigen hundert Jahren. Sie haben sich einen speziellen Code ausgedacht.  
Na, was meint ihr, könnt ihr diese Nachricht lesen?

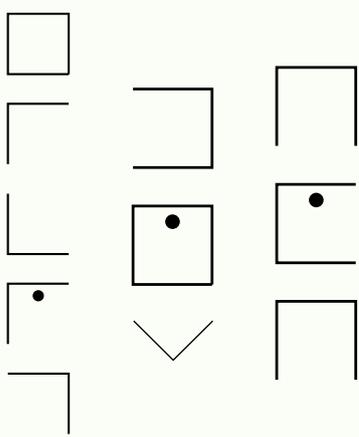


Ich habe eurem Lehrer auch eine kleine Hilfe mitgeschickt.

Vielleicht könnt ihr ja nachher sogar selber wie die Freimaurer Geheimnachrichten schreiben.

Viel Spaß!  
Eure Alice

KV Brief3




KV Nachricht im Freimaurercode

**Der Code der Freimaurer**

Hier siehst du den Code der Freimaurer.  
Der Code ist das Muster, das um den Buchstaben herum entsteht.

A	D	G	J	M	P	S	W
B	E	H	K	N	Q	T	X
C	F	I	L	O	R	U	Y
						V	Z

Man nennt diese Darstellung eine **Code-Tabelle**.  
Mit ihr kannst du alle Buchstaben als Code schreiben.

A	B	C	D	E	F
G	H	I	J	K	L
M	N	O	P	Q	R
S	T	U	V	W	X
Y	Z				

**Aufgabe**  
Kannst du den Freimaurercode decodieren?  
Finde in der Code-Tabelle den passenden Buchstaben neben dem Bild.  
Schreibe deine Lösung darunter.



KV Freimaurercode-Tabelle

**Geheime Nachrichten schreiben wie die Freimaurer**

**Aufgabe**  
Arbeite mit deinem Partner.

- Schreibe dein Wort in das obere Feld.
- Codiere den Satz mit der Code-Tabelle der Freimaurer. So erhältst du deine geheime Nachricht.
- Schneide die geheime Nachricht an den gestrichelten Linien aus und übergib sie am Meeting-Point einem anderen Team. Lasse dir auch ihre Geheimnachricht geben.
- Decodiere die Geheimnachricht der anderen Kinder.

**Dein Wort:**

**Deine geheime Nachricht:**

**Die decodierte Geheimnachricht der anderen Kinder:**

AB Codieren wie ein Freimaurer

# Codierung

optional: Codes sind überall

Unterrichtseinheit 2a

## Codes sind überall

**Hinweis**

Diese Unterrichtseinheit ist optional und stellt eine Ergänzung zum Thema Codierung dar.

**Ziele und Kompetenzen**

Die Schülerinnen und Schüler können...



fachsprachlich korrekt über Codierung kommunizieren und argumentieren.



im Alltag auftretende Formen von Codierung beschreiben.

**Zeit**

1–2 Doppelstunde

**Lernvoraussetzungen**

Gruppenarbeit



Lernplakate



Präsentieren

**Empfohlene Vorgehensweise**

- ▶ Lassen Sie die Schülerinnen und Schüler wiederholen, was sie über den Freimaurercode herausgefunden haben.
- ▶ Fragen Sie die Schülerinnen und Schüler, ob sie auch andere Codes kennen. Ermöglichen Sie ggf. eine kurze Murmelrunde zum Austausch.
- ▶ Sammeln Sie weitere Erscheinungen von Codierung im Alltag und notieren Sie sie auf Kärtchen für ein Cluster.
- ▶ Formulieren Sie gemeinsam mit den Schülerinnen und Schülern eine Definition für Codes.
- ▶ Beispiel: *Ein Code stellt eine Nachricht anders dar. Der Empfänger muss den Code auch kennen, damit er die Nachricht lesen kann. Codes sind nicht geheim, jeder kann sie lernen. Es gibt Code-Tabellen. Das Schreiben einer solchen geheimen Nachricht nennen wir Codieren, das Lesen nennen wir Decodieren.*
- ▶ Erläutern Sie den Schülerinnen und Schülern den Aufbau der Kopiervorlagen zu weiteren Codes.
- ▶ Besprechen Sie die Aufgabe: Zur Erstellung eines gemeinsamen Klassenplakats über Codes finden sich die Schülerinnen und Schüler in Gruppen zusammen und wählen einen Code aus.
- ▶ Die Schülerinnen und Schüler lesen den kurzen Text, tauschen sich über den Code aus und erstellen ihren Teil für ein Plakat: Überschrift, Text, Bild(er).
- ▶ Kleben Sie mit den Schülerinnen und Schülern ihre fertigen Bausteine auf das Klassenplakat.
- ▶ Lassen Sie die Schülerinnen und Schüler im Plenum kurz präsentieren, welchen Code sie bearbeitet haben und was sie herausgefunden haben.

## Codes sind überall

**Empfohlene  
Vorgehens-  
weise**

- ▶ Geben Sie für die nächste Stunde den Ausblick, dass es mit Alice, Bob und Eve und den geheimen Nachrichten weitergeht.
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

**Material-  
überblick**

 KV Codes sind überall

**Zusätzliches  
Material**

-  Tonpapier
-  Schere und Kleber
-  Stifte
-  Wandzeitung

# Materialübersicht: Codes sind überall (Auszug)

**Codes sind überall!**  
Methode Klassenplakat

**Aufgabe**  
😊😊😊 Arbeit in Gruppen.  
Erstellt einen Beitrag zu einem gemeinsamen Klassenplakat über Codes.

- 👁️ Lest den Text und schaut euch das Bild an.  
Warum handelt es sich um einen Code?
- ✋ Bereitet einen Text für das Lernplakat vor. Ihr könnt den Text ausschneiden, abschreiben oder euch einen eigenen Text überlegen.
- ✂️ Schneidet das Bild aus und gestaltet eine Überschrift.
- 🗨️ Überlegt euch, was ihr den anderen Kindern vorstellt und übt eure Präsentation kurz ein.
- 👄 Stellt den anderen Kindern eure Codierung vor und klebt sie mit auf das Klassenplakat.

**Codes sind überall!**  
Methode Klassenplakat

**Aufgabe**  
😊😊😊 Arbeit in Gruppen.  
Erstellt einen Beitrag zu einem gemeinsamen Klassenplakat über Codes.

- 👁️ Lest den Text und schaut euch das Bild an.  
Warum handelt es sich um einen Code?
- ✋ Bereitet einen Text für das Lernplakat vor. Ihr könnt den Text ausschneiden, abschreiben oder euch einen eigenen Text überlegen.
- ✂️ Schneidet das Bild aus und gestaltet eine Überschrift.
- 🗨️ Überlegt euch, was ihr den anderen Kindern vorstellt und übt eure Präsentation kurz ein.
- 👄 Stellt den anderen Kindern eure Codierung vor und klebt sie mit auf das Klassenplakat.

AB Arbeitsauftrag

**Codes sind überall**

📌 Unser Code:

### Die Anlauttabelle

In der Grundschule lernen viele Kinder Lesen und Schreiben mit Anlauttabelle. Dabei wird jedem Buchstaben ein Bild zugeordnet, das mit dem gleichen Laut anfängt.  
Wenn man ein U schreiben will, sucht man das Bild mit dem U am Anfang – Ufo. Darunter sieht man, wie das U geschrieben wird.  
Die Buchstaben und das Bild sind also ein Code für den Laut.



AB Code: Anlaute

**Codes sind überall**

📌 Unser Code:

### Musik-Noten

Auch Noten für ein Musikstück sind ein Code.  
Die Linie, auf der die Note steht, zeigt an, welcher Ton gespielt werden muss. Ein leerer Notenkopf ist eine halbe Note, sie wird länger gespielt als ein voller Notenkopf. Auf dem Bild unten siehst du ein bekanntes amerikanisches Volkslied. Man kann Noten decodieren, wenn man sie auf einem Instrument spielt.

**Amazing Grace**

NEW BRITAIN C.M. VIRGINIA HARMONY harm. Edwin O. Excell



JOHN NEWTON VS. 5 ATTR. JOHN REESE

AB Code: Musik-Noten

**Codes sind überall**

📌 Unser Code:

### Verkehrsschilder

Verkehrsschilder sind auch ein Code.  
Es gibt Zeichen die dir etwas verbieten, um dich zu beschützen.  
Einige Zeichen zeigen dir, wo du gehen oder fahren darfst.  
Manchmal ist die Farbe auch ein Code.  
Als Kind lernst du einige Verkehrszeichen beim Radfahrtraining kennen.  
Wenn du den Code kennst, darfst du auch auf der Straße Fahrrad fahren.  
Menschen, die den Code nicht kennen, wissen nicht, was ein rotes Dreieck auf dem Kopf mit weiß innen drin bedeutet.  
Du weißt: **Vorfahrt achten!**



AB Code: Verkehrsschilder

# Transposition

## Die Skytale

Unterrichtseinheit 3

## Die Skytale



## Ziele und Kompetenzen

Die Schülerinnen und Schüler...

-  lernen das Verschlüsselungsverfahren der Transposition am Beispiel Skytale kennen.
-  können die einzelnen Schritte des Algorithmus zur Ver- und Entschlüsselung mit Skytalen beschreiben und anwenden.
-  kennen die Begriffe *Skytale*, *Algorithmus*, *Verschlüsseln*, *Entschlüsseln*.

## Zeit

 1 Doppelstunde

## Lernvoraussetzungen

-  keine zusätzlichen Lernvoraussetzungen nötig

## Empfohlene Vorgehensweise

- ▶ Lesen Sie den Kindern Brief 4 und die Geschichte zur Skytale vor.
- ▶ Teilen Sie den ersten halben Klassensatz Skytale mit den dazu passend vorbereiteten und beschriebenen Papierstreifen an die Schülerinnen und Schüler aus. *Bitte beachten Sie die Hinweise auf der übernächsten Seite.*
- ▶ Lassen Sie die Schülerinnen und Schüler in Partnerarbeit entdeckend herausfinden, wie die verschlüsselten Nachrichten entschlüsselt werden können. Welche Handlung muss durchgeführt werden, um die Klarnachricht zu erhalten?
- ▶ Besprechen Sie im Plenum, wie die Schülerinnen und Schüler vorgehen mussten, um die Nachricht zu entschlüsseln.  
Führen Sie hierbei den Begriff *Algorithmus* ein und umschreiben Sie ihn als *Anleitung für eine Tätigkeit oder Handlung*.
- ▶ Geben Sie den Schülerinnen und Schülern den nächsten Satz vorbereitete Nachrichten, der diesmal zu einem anderen Durchmesser passt und stellen Sie ihnen weitere Stäbe mit unterschiedlichen Durchmessern und Formen zur Verfügung, so dass sie den passenden Stab für die verschlüsselte Nachricht finden müssen.  
Die Schülerinnen und Schüler werden merken, dass Papierstreifen (*Verschlüsselte Nachricht*) und Holzstab bzw. Skytale (*Schlüssel*) zusammenpassen müssen, damit man die Nachricht lesen kann (*Entschlüsseln des Klartextes*).
- ▶ Lassen Sie die Schülerinnen und Schüler begründen, warum man die verschlüsselte Nachricht nicht lesen kann, wenn die falsche Skytale (der falsche *Schlüssel*) verwendet wird. Thematisieren Sie auch, wie die Verschlüsselung überhaupt funktioniert. Impulsfragen können sein:
  - Sind Buchstaben dazugekommen oder verschwunden?
  - Haben die Buchstaben noch die gleiche Reihenfolge?
  - Warum sind die Buchstaben vertauscht?
  - Kann man die Nachricht auch ohne Holzstab entschlüsseln?

## Die Skytale

**Empfohlene Vorgehensweise**

- ▶ Überlegen Sie mit den Schülerinnen und Schülern gemeinsam, wie man eine Nachricht sinnvoll verschlüsseln, austauschen und entschlüsseln kann.
- ▶ Lassen Sie die Schülerinnen und Schüler auf vorbereiteten Papierstreifen eigene Nachrichten verschlüsseln und austauschen.
- ▶ Lesen Sie den Schülerinnen und Schülern Brief 5 vor und überlegen Sie als Ergebnis-sicherung im Plenum, wie Eve die verschlüsselte Nachricht lesen konnte.
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

**Material-überblick**

- KV Brief 4
- KV Die Geschichte der Skytale
- KV Brief 5
- Optionaler Sachtext – »Warum ist Verschlüsselung wichtig?«

**Zusätzliches Material**

*Die Durchmesserangaben dienen lediglich als Anhaltspunkt. Eine genaue Angabe von cm ist nicht sinnvoll. Bitte beachten Sie die Hinweise zur Materialerstellung auf der nächsten Seite.*

- Ein halber Klassensatz Stäbe mit einem Durchmesser von X cm.
- Stäbe im Durchmesser von Y cm und weitere Stäbe in verschiedenen Formen und Stärken.
- Verschlüsselte Nachrichten für die Skytale
- Blankopapierstreifen für einen Klassensatz
- Wandzeitung

## Hinweis zur Materialerstellung: Skytale



Nachfolgend wird erläutert, wie das Arbeitsmaterial zur Skytale erstellt werden kann.

Für die Skytale benötigt man grundsätzlich einen Stab und einen Papierstreifen. Der Stab kann alles Erdenkliche sein: ein Bleistift bzw. Fineliner, ein Besenstiel, ein Plastikrohr, ein Rundholz, ein Kantholz, etc.

Ihrer Fantasie sind hier keine Grenzen gesetzt. Die Auswahl des Materials hängt daher von der Lerngruppe ab. Erfahrungen haben gezeigt, dass Stifte als Skytale ggf. zu klein für die Schüler sind. Wenn Sie Holzstäbe nutzen, können Sie diese am Ende einkerben. Dort können die Schülerinnen und Schüler den Anfang des Papierstreifens einstecken und ihn so leichter auf den Stab wickeln.

### So erstellen Sie geheime Nachrichten mit der Skytale

Schneiden Sie für die Papierstreifen zunächst ein Din A4-Blatt längs auf 8 gleichmäßige Streifen zu. Für die Anleitung wurde dickeres Papier (80g/cm<sup>2</sup>) verwendet.



Treffen Sie eine Auswahl für die passenden Stäbe: Sie benötigen **2 halbe Klassensätze mit jeweils dem selben Durchmesser** (z. B. je 15 Stäbe mit 2 cm und je 15 Stäbe mit 4 cm). Sie sind frei in der Wahl der Durchmesser.



Sie benötigen **weitere Stäbe in unterschiedlichen Durchmessern** und Formen (z. B. Besenstiel, Tafellineal, Stifte, Spazierstock, Äste, Kanthölzer, etc.).



## Hinweis zur Materialerstellung: Skytale



Wickeln Sie einen Papierstreifen um einen Stab.



Notieren Sie auf dem Stab Ihre geheime Nachricht.



Wickeln Sie den Streifen wieder ab.



Sie können nach Bedarf diese selbst erstellte Vorlage vervielfältigen und zur Verwendung durch die Schülerinnen und Schüler vorbereiten.

Auf der Internetseite der Didaktik der Informatik der Universität Wuppertal finden Sie ein *Werkzeug* (engl.: *Tool*) zur automatisierten Erstellung von Skytale-Nachrichten. Zur Verwendung mittels Kommandozeile benötigen Sie  $\text{\LaTeX}$  und Python.

» Philipp Rumm, Hrsg. *Tool zur Erstellung von Skytale-Nachrichtenstreifen unter Verwendung von  $\text{\LaTeX}$  und Python*. 2018. URL: <https://uni-w.de/zo> (besucht am 09. 03. 2019)

# Materialübersicht: Die Skytale

**Brief 4** ☺☹☹☹



Hallo Kinder,

Ihr habt neulich heraus gefunden, dass der Freimaurercode gar nicht so sicher war, wenn man wusste, wie es geht.  
Danke für eure Hilfe!

Alice und ich haben überlegt, wie wir unsere Nachrichten noch geheim halten können.

Wir haben etwas gefunden und eurer Lehrerin mitgegeben.

Soviel sei schon einmal gesagt: Es ist eine richtige Verschlüsselung, wie sie die alten Griechen auch schon benutzt haben, um Krieg zu führen und Schlachten zu gewinnen.

Ich bin gespannt, was ihr davon haltet!

Viel Spaß,  
Euer Bob

KV Brief 4

**Brief 5** ☺☹☹☹



Hallo Kinder,  
ich bin's, eure Alice.

Ich habe das Gefühl, dass ihr mit den Skytalen euren Spaß hattet. Toll wie ihr herausgefunden habt, wie man die Holzstäbe benutzt, um Nachrichten zu verschlüsseln und zu entschlüsseln.

Aber wisst ihr was? Eve hat es auch wieder hinbekommen! Jetzt kennt sie Bobs und meine Nachrichten. Wie hat sie das nur geschafft? Habt ihr eine Idee? Und noch wichtiger: was kann man dagegen tun? Ich freue mich über eure Hilfe!

Viele Grüße,  
Eure Alice

KV Brief 5

**Die Geschichte der Skytale** ☺☹☹☹

Die Skytale ist ein Paar aus zwei Zylindern mit dem gleichen Durchmesser. Sie wurden zur Zeit der alten Griechen vor ca. 2500 Jahren benutzt, um geheime Nachrichten zu übermitteln. Meistens ging es dabei um Krieg und Eroberungen. So konnte der spartanische General Lysander während einer großen Schlacht einen Angriff abwehren, weil er eine Vorwarnung durch eine Skytale erhalten hatte. Sein Spion bei den Gegnern schrieb ihm die Angriffspläne auf und niemand außer dem General konnte sie lesen.

**Aber wie benutzt man die Skytale?**



KV Geschichte der Skytale

# Substitution

## Caesar-Verschlüsselung mit Code-Scheibe

### Unterrichtseinheit 4

## Caesar-Verschlüsselung mit Code-Scheibe

**Ziele und Kompetenzen**

Die Schülerinnen und Schüler...

-  lernen den Umgang mit einem Verfahren zur Substitution am Beispiel der *Caesar-Code Scheibe* kennen.
-  können den zugrunde liegenden Algorithmus der Verschlüsselung ausführen.

**Zeit**

 1 Doppelstunde

**Empfohlene Vorgehensweise**

- ▶ Bereiten Sie mit der Kopiervorlage für die Code-Scheiben diese zweifarbig vor. Nutzen Sie für die große Scheibe rotes Tonpapier und für die kleine Scheibe grünes Tonpapier.
- ▶ Kopieren Sie ggf. eine Code-Scheibe in groß zur Demonstration und händigen Sie jedem Kind eine fertig vorbereitete Code-Scheibe aus.
- ▶ Lesen Sie Brief 6 vor. Schreiben Sie am Ende die verschlüsselten Wörter an die Tafel.
- ▶ Geben Sie den Schülerinnen und Schülern die Code-Zahl vor und lassen Sie sie versuchen, die Wörter gemeinsam zu entschlüsseln (Lösung: »Hallo Kinder«).
- ▶ Verschlüsseln Sie danach mit der Klasse die gleichen Wörter mit einer anderen Code-Zahl.
- ▶ Lassen Sie die Schülerinnen und Schüler mit der Vorlage eigene Wörter verschlüsseln und durch Weitergabe der Code-Zahl von zufälligen Kindern am Meeting-Point entschlüsseln.
- ▶ Integrieren Sie am Ende im Gespräch mit den Schülerinnen und Schülern, an welchen Stellen Lernhürden auftraten und lassen Sie bewerten, wie sicher dieses Verfahren ist.
- ▶ Thematisieren Sie den praktischen Einsatz des Verfahrens im Alltag.
- ▶ *Hinweis: Zur intensiveren Auseinandersetzung mit dem Verfahren können die Kinder die Scheiben auch selbst erstellen.*
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

**Material-überblick**

-  KV Brief 6
-  KV Caesar-Scheibe groß (rot kopieren)
-  KV Caesar-Scheibe klein (grün kopieren)
-  AB Nachrichten mit Caesar verschlüsseln

**Zusätzliches Material**

-  Wandzeitung

## Hinweis zur Materialerstellung: Caesar-Scheibe



Sie benötigen für jedes Kind je eine große und eine kleine Scheibe sowie eine Musterbeutelklammer.

1. Achten Sie darauf, dass Sie beim Ausdrucken die Größe des Dokuments beibehalten (100% Druck).  
Die Original-Durchmesser sind: großes Rad: 11cm – kleines Rad: 8,6cm.  
Drucken Sie unbedingt auf *farbiges* Papier, damit die Kinder die Bedeutungen der Scheiben unterscheiden können.
2. Schneiden Sie beide Scheiben aus und laminieren Sie diese.
3. Vergessen Sie nicht, in der kleinen Scheibe das graue Code-Fenster (X) auszuschneiden.
4. Verbinden Sie beide Scheiben mit einer Musterbeutelklammer.

### Handhabung Verschlüsselung

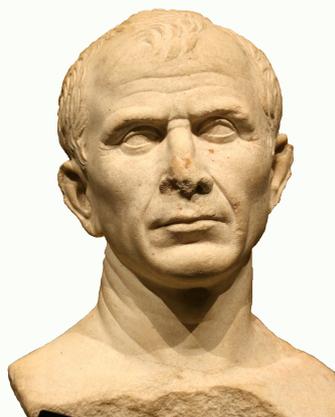
Kleine Scheibe auf die Code-Zahl einstellen.

Innen den Buchstaben suchen und außen den Geheimbuchstaben ablesen.

### Handhabung Entschlüsselung

Kleine Scheibe auf die Code-Zahl einstellen.

Außen den Buchstaben suchen und innen den Klarbuchstaben ablesen.



# Materialübersicht: Die Caesar-Verschlüsselung mit Code-Scheibe

**Brief 6**



Hallo Kinder,  
ich bin's wieder, Alice.

Die Skytale letztes Mal waren schon *ziemlich* cool!  
Aber ihr habt ja herausgefunden, dass es auch *echt* einfach ist, sie zu knacken, wenn man ein bisschen herumprobiert.

Heute bekommt ihr den letzten Brief von mir.  
Warum? Ganz einfach! Ab heute seid ihr richtige Verschlüsselungs-Profis!

Zum Abschluss haben Bob und ich euch noch etwas Spannendes mitgebracht.  
Die Verschlüsselung, die ihr heute kennen lernt, gibt es auch schon sehr lange. Erfunden wurde sie von einem der wichtigsten Menschen der Geschichte – Caesar.  
Er war ein großer Herrscher der Römer.  
Weil er so wichtig war, musste auch er seine Nachrichten immer gut geheim halten.  
Dafür hat er sich eine Scheibe zur Verschlüsselung ausgedacht.  
Es war zwar total einfach die Nachricht zu entschlüsseln, wenn man wusste, wie es ging.  
Aber wenn man es nicht wusste, konnte man sich daran auch die Zähne aus beißen.

Da ihr ja schon Profis seid, sag ich nur so viel:  
**Die Zahl ist 3.** Dazu braucht ihr noch die **Code-Scheibe**.  
Na, könnt ihr die Nachricht entschlüsseln?

Viele Grüße, *Eure Alice*

KV Brief 6

**Eine geheime Nachricht von Alice**

**Aufgabe**  
Arbeite mit deinem Partner.

- Benutzt die **Code-Scheibe** mit der Zahl **3**.  
Untersucht dafür die Code-Scheibe und stellt sie richtig ein.
- Versucht gemeinsam, die verschlüsselte Nachricht zu **entschlüsseln**.  
Schreibt euer Ergebnis in das richtige Feld.

**Alice' geheime Nachricht**

K D O O R   N L Q G H U

**Unser Ergebnis der Entschlüsselung:**

- Verschlüsselt nun selber Nachrichten.  
Welche Zahl habt ihr für die Code-Scheibe benutzt?
- Kontrolliert eure Verschlüsselung.  
Gebt euren Abschnitt am Meeting-Point an ein anderes Team weiter.

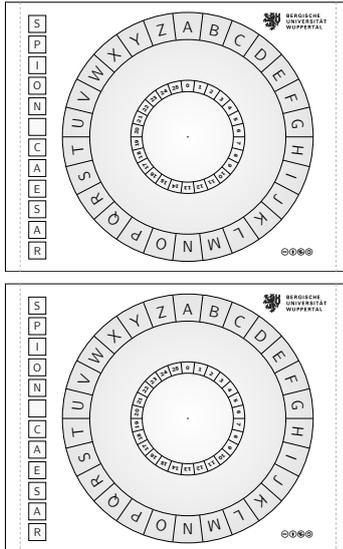
**Geheimnachricht**

Zahl: \_\_\_\_\_  
Nachricht: \_\_\_\_\_

**Ergebnis der Entschlüsselung:**

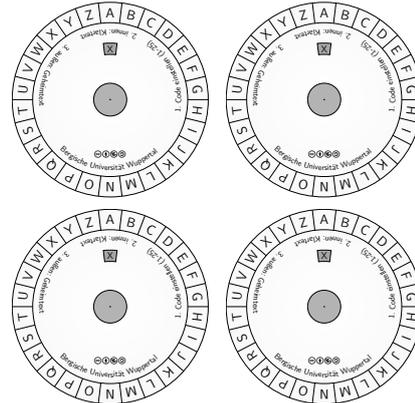
AB Nachricht ver-/entschlüsseln mit Caesar

Diese Seite auf roten Karton kopieren.



KV Codescheibe für die Caesar-Verschlüsselung

Diese Seite auf grünen Karton kopieren.



KV Codescheibe für die Caesar-Verschlüsselung

# Verschlüsselung im Alltag

## Messenger und Co.

Unterrichtseinheit 5

## Verschlüsselung im Alltag

**Ziele und Kompetenzen**

Die Schülerinnen und Schüler...

- 🎓 übertragen ihr Wissen über Kryptologie auf Verschlüsselungsszenarien im Alltag.
- 🎓 können Angriffsszenarien für alltägliche Verschlüsselungsverfahren angeben.

**Zeit**

1 Einzelstunde

**Empfohlene Vorgehensweise**

- ▶ Bereiten Sie die KV Bildkarten für alltägliche Verschlüsselungsverfahren für den Gesprächskreis oder eine Plenumsphase mit Tafel vor.
- ▶ Klären Sie mit den Schülerinnen und Schülern die Bedeutungen der Bildkarten und erläutern Sie ggf. die dazugehörigen Verschlüsselungsszenarien skizzenartig.
- ▶ Ergänzen Sie die Wortkarten zu den verschiedenen Angriffsszenarien. Lesen Sie dazu den Beschreibungstext vor und erarbeiten Sie mit den Schülerinnen und Schülern das Vorgehen bei den Angriffen.
- ▶ *Nutzen Sie dabei die Fachsprache und besprechen Sie mit den Schülerinnen und Schülern die Bedeutungen der Fachbegriffe. Sammeln Sie die Fachbegriffe ggf. an einer Wandzeitung.*

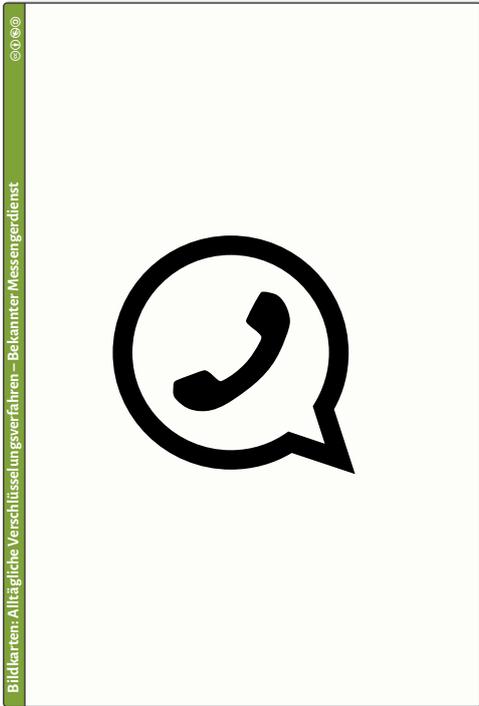
**Material-überblick**

- KV Bildkarten: Alltägliche Verschlüsselungsverfahren
- KV Angriffsszenarien

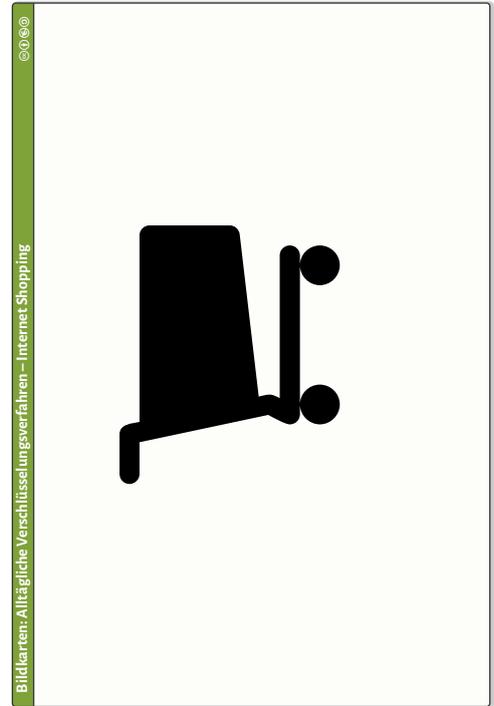
**Zusätzliches Material**

- Wandzeitung

# Materialübersicht: Verschlüsselung im Alltag (Auszug)



KV Messengerdienst



KV Internet Shopping

Abgriffsszenario – Brute-Force



**Brute-Force** bedeutet auf Englisch *rohe Gewalt*.  
**Eve** versucht die Verschlüsselung zu knacken.  
 Das macht **Eve** solange, bis sie erfolgreich ist.

---

**Eve** probiert z. B. ganz lange alle Passwörter aus, die ihr einfallen.  
 Irgendwann ist **Eve** dann vielleicht erfolgreich.

KV Angriffsszenario Brute-Force

Abgriffsszenario – Man-in-the-Middle



**Man-in-the-Middle** bedeutet auf Englisch *Mann in der Mitte* oder *Mittelsmann*.

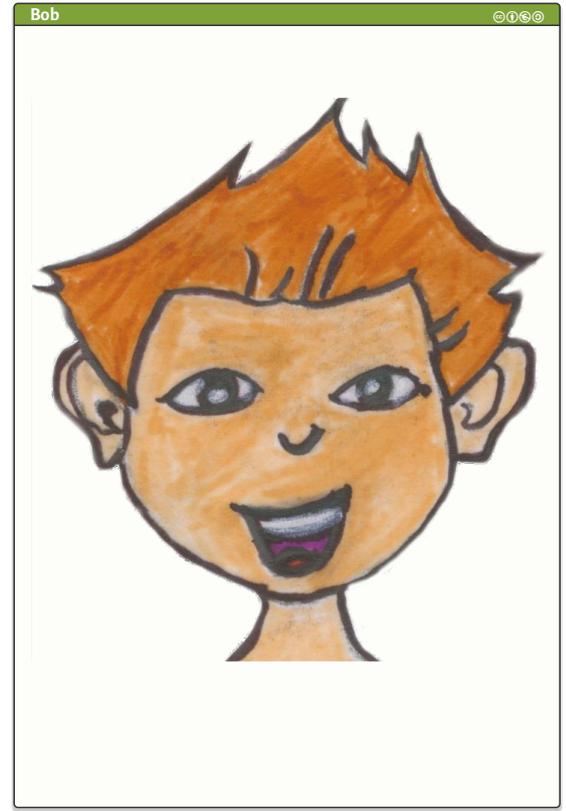
**Eve** versucht, die Nachricht abzufangen, die eigentlich für Bob bestimmt war.  
 Wenn **Eve** die Nachricht abgefangen hat, kann **Eve** sie verändern und dann an Bob weitergeben.  
 Bob denkt, dass er eine unveränderte Nachricht von Alice bekommt.

KV Angriffsszenario Man-in-the-Middle

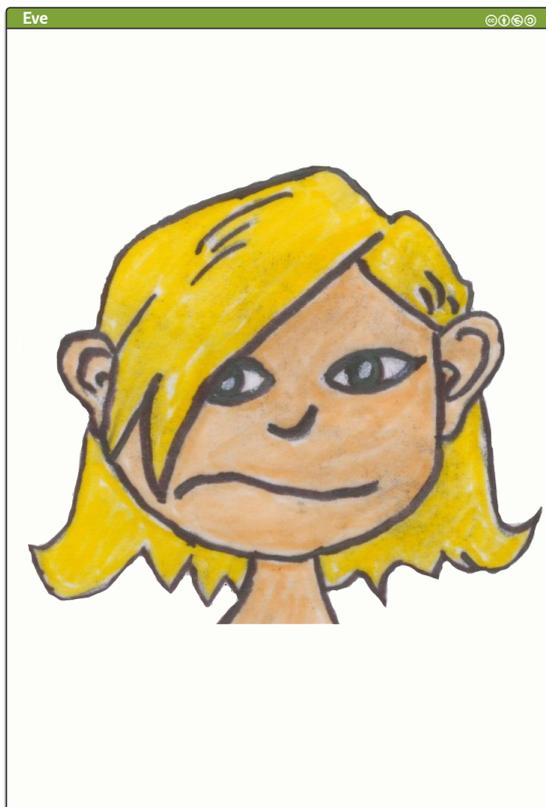
## Materialübersicht: Weitere Kopiervorlagen (Auszug)



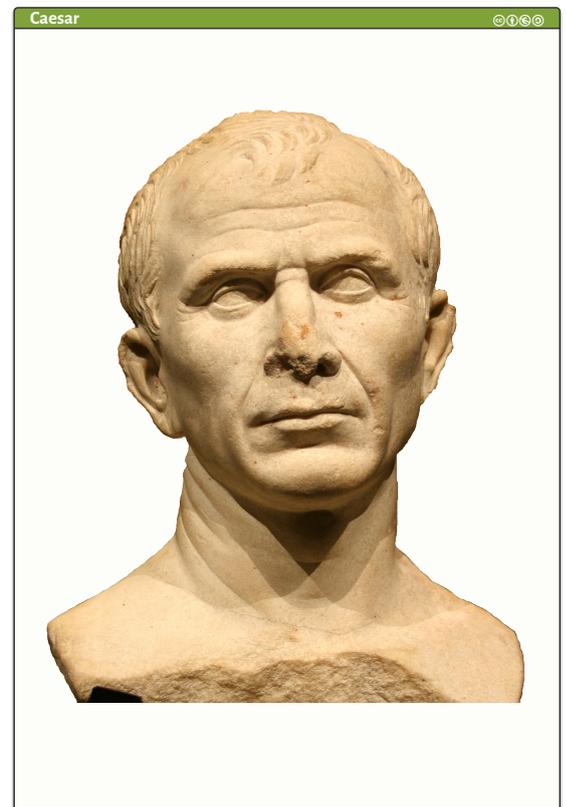
KV Alice



KV Bob



KV Eve



KV Caesar

## Materialübersicht: Wortspeicherkarten (Auszug)

**Algorithmus**

Ein Algorithmus ist eine Abfolge von Anweisungen, die Schritt für Schritt ausgeführt werden.  
 Ein **Rezept** oder eine **Spielanleitung** sind Algorithmen.  
 Computer arbeiten auch mit Algorithmen.

KV Wortspeicher»Algorithmus«

**Codierung**

Vereinbarung von einem Zeichen für einen bestimmten Inhalt.  
 Blindenschrift ist eine Codierung.  
 Verkehrsschilder sind Codierungen.  
 Die Anlauttabelle ist eine Codierung.

KV Wortspeicher»Codierung«

**Informatik**

**Informatik = Information + Automatik.**  
 Informatik ist die Wissenschaft der automatischen Verarbeitung von Information.

KV Wortspeicher»Informatik«

**Steganographie**

Das Verbergen einer Botschaft an einem sicheren Ort, so dass sie nur vom Empfänger gefunden werden kann.  
 Für alle anderen Menschen ist die Nachricht also unsichtbar.  
 Geheimstifte und Geheimtinte sind ein Beispiel für Steganographie.

KV Wortspeicher»Steganographie«

## 8 Weiterführende Materialien – Modul Kryptologie

### Literatur für die Kinderhand

- Rudolf Kippenhahn. *Streng geheim! Wie man Botschaften verschlüsselt und Zahlencodes knackt*. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag, 2003. ISBN: 3499211645
- Monika Schlitzer. *Memo – Wissen Entdecken: Computer*. Dorling Kindersley Verlag, 2012. ISBN: 3831020477, S. 50
- Michael Kohlhammer. *Geolino: Spionage: Knifflige Codes und geheime Fälle*. Franckh Kosmos Verlag, 2011. ISBN: 978-3440128244
- Martin Verg. *GEOLino extra 34-2012 – Spione und Detektive*. Gruner + Jahr, 2012. ISBN: 978-3652001199
- Camille Denton Gillian und Hallinan. *Memo – Wissen: Spione*. Dorling Kindersley Verlag, 2011. ISBN: 3831019061
- Michael Hielscher. *Mal-den-Code.de*. 9. Mai 2015. URL: <https://t1p.de/vwzy> (besucht am 09. 03. 2019)
- Claire Calmet, Mathieu Hirtzig und David Wilgenbus. *1, 2, 3 ...kodiert, programmiert, verschlüsselt! – Informatik in der Grundschule*. 17. Jan. 2017. URL: <https://t1p.de/1v2h> (besucht am 09. 03. 2019)

### Vertiefende Literatur für Lehrpersonen

- Didaktik der Informatik an der Bergischen Universität Wuppertal, Hrsg. *Internetpräsenz der Ddl für den Primarbereich*. 2018. URL: <https://uni-w.de/zn> (besucht am 09. 03. 2019)
- Simon Singh. *Codes. Die Kunst der Verschlüsselung*. 2014. ISBN: 9783423621670
- Jens Gallenbacher. *Ausstellung: Abenteuer Informatik – Informatik begreifen*. Darmstadt, 2008. URL: <https://t1p.de/yqik> (besucht am 09. 03. 2019)
- Ernestine Bischof und Roland Mittermeir. *Informatik erLeben. Beispiele für schülerinnen- und schüleraktivierenden Informatikunterricht*. 2010. URL: <https://t1p.de/5gzo> (besucht am 09. 03. 2019)

## 9 Verwendungsnachweis – Modul Kryptologie

### Abbildungsverzeichnis

	Logo des Projekts Informatik an Grundschulen (IaG). Fortlaufende Verwendung. . . . .	KR i
	Logo des Ministeriums für Schule und Bildung Nordrhein-Westfalen. . . . .	KR i
	Logo der Bergischen Universität Wuppertal. . . . .	KR i
	Logo der Didaktik der Informatik an der Bergischen Universität Wuppertal. . . . .	KR i
	Logo des Moduls Kryptologie im Projekt Informatik an Grundschulen (IaG). . . . .	KR ii
1	Modellierungskreis der Informatik (nach Humbert, 2006, S. 14). . . . .	KR 04
2	Die Fachgebiete der Informatik (vgl. Humbert, 2018) – Anpassung durch Fricke. . . . .	KR 06
3	Beispiel für eine Freimaurer-Chiffre. Eigene Abbildung. . . . .	KR 11
4	Beispiel für eine Codierung mit Freimaurer-Chiffre: »Alice und Bob«. Eigene Abbildung. . . . .	KR 11
5	Skytale mit Lederriemen (Luringen, 2007) – Abbildung steht unter CC-BY-SA Lizenz. . . . .	KR 12
6	Gartenzaun. Eigene Abbildung. . . . .	KR 13
7	Transposition mit Schablone (vgl. Müller, 2012) – Abbildung steht unter CC-BY-NC-SA Lizenz. . . . .	KR 13
8	Caesar-Chiffre mit Verschiebung um drei Stellen. Eigene Abbildung. . . . .	KR 14
9	Caesar-Chiffre mit Scheiben (vgl. Müller, 2012) – Abbildung steht unter CC-BY-NC-SA Lizenz. . . . .	KR 14
10	Caesar-Chiffre mit Codewort und anschließendem Auffüllen. Eigene Abbildung. . . . .	KR 15

## Abbildungen im Unterrichtsmaterial – Modul Kryptologie

(In alphabetischer Reihenfolge)

Abbildung	Quelle
Anlauttabelle	Wolfram Esser. <i>Anlauttabelle klein - Ausschnitt Buchstabe U.png</i> . 2009. URL: <a href="https://t1p.de/9380">https://t1p.de/9380</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.
Anleitung Skytale	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Alice	Dorothee Müller. <i>Alice.png</i> . In: <i>SpionCamp – Stationen</i> . 2012. URL: <a href="https://uni-w.de/152">https://uni-w.de/152</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz. Anpassung durch Fricke.
Bildkarten: Verschlüsselungen aus der Lebenswirklichkeit	Xavier Danaux und Dave Gandy. <i>Fontawesome Package für L<sup>A</sup>T<sub>E</sub>X</i> . 2016. Abbildungen stehen unter SIL Open Font License.
Blindenschrift	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Bob	Dorothee Müller. <i>Bob.png</i> . In: <i>SpionCamp – Stationen</i> . 2012. URL: <a href="https://uni-w.de/152">https://uni-w.de/152</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz. Anpassung durch Fricke.
Caesar Büste	Wikipedia-Nutzer DanieleDF1995. <i>Caesar Büste</i> . 2010. URL: <a href="https://t1p.de/q6dg">https://t1p.de/q6dg</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.
Caesar-Scheibe groß	Dorothee Müller. <i>caesar-big.pdf</i> . In: <i>SpionCamp – Stationen</i> . 2012. URL: <a href="https://uni-w.de/152">https://uni-w.de/152</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.
Caesar-Scheibe klein	Dorothee Müller. <i>caesar-small.pdf</i> . In: <i>SpionCamp – Stationen</i> . 2012. URL: <a href="https://uni-w.de/152">https://uni-w.de/152</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.
DNS	Openclipart-Nutzer Anonymous. <i>Double Helix (DNA)</i> . 2006. URL: <a href="https://t1p.de/qb7k">https://t1p.de/qb7k</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
EAN-Code	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Eve	Dorothee Müller. <i>Eve.png</i> . In: <i>SpionCamp – Stationen</i> . 2012. URL: <a href="https://uni-w.de/152">https://uni-w.de/152</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz. Anpassung durch Fricke.

Fortsetzung auf der nächsten Seite...

Abbildung	Quelle
Freimaurer-Chiffren	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Freimaurer-Werkzeug	Wikipedia-Nutzer MesserWoland. <i>compasses.svg</i> . 2006. URL: <a href="https://t1p.de/rgsy">https://t1p.de/rgsy</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.
Gebärdensprache	Wikipedia-Nutzer Ds13 (English Wikipedia). <i>Asl alphabet gallaudet.png</i> . 2004. URL: <a href="https://t1p.de/u0nw">https://t1p.de/u0nw</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Meeting-Point	Wikipedia-Nutzer Xavax. <i>Sammelpunkt.svg</i> . 2007. URL: <a href="https://t1p.de/uuo2">https://t1p.de/uuo2</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz. Abgewandelt durch Fricke.
Morsecode	V. F. Camp R. T. Snodgrass. <i>International Morse Code.svg</i> . 2008. URL: <a href="https://t1p.de/au8l">https://t1p.de/au8l</a> (besucht am 09. 03. 2019). Die Abbildung steht unter CC0 Lizenz.
Musik-Noten	Brian Ammon. <i>AmazingGrace.svg</i> . 2012. URL: <a href="https://t1p.de/0bgt">https://t1p.de/0bgt</a> (besucht am 09. 03. 2019). Die Abbildung steht unter CC0 Lizenz.
QR-Code	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Spartanischer Krieger	Wikipedia-Nutzer SAWg3rd. <i>SpartanWarriorAgoge.jpg</i> . 2004. URL: <a href="https://t1p.de/xkra">https://t1p.de/xkra</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Sklavenkopf	Pixabay.com. <i>Glatze</i> . 2011. URL: <a href="https://t1p.de/j0rw">https://t1p.de/j0rw</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Steganographie Materialerstellung	Didaktik der Informatik an der Bergischen Universität Wuppertal.
Verkehrsschild Ampel	Pixabay.com. <i>Ampel</i> . 2011. URL: <a href="https://t1p.de/h8yy">https://t1p.de/h8yy</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Verkehrsschild Fußgängerweg	Pixabay.com. <i>Fußgängerweg</i> . 2011. URL: <a href="https://t1p.de/tiku">https://t1p.de/tiku</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Verkehrsschild Vorfahrt achten	Pixabay.com. <i>Vorfahrt achten</i> . 2011. URL: <a href="https://t1p.de/epsp">https://t1p.de/epsp</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Wachstafel	Wikipedia-Nutzer Sippel2707. <i>Wachstafel.jpg</i> . 2011. URL: <a href="https://t1p.de/vomx">https://t1p.de/vomx</a> (besucht am 09. 03. 2019). Abbildung steht unter CC-BY-SA Lizenz.

Fortsetzung auf der nächsten Seite...

Abbildung	Quelle
	<i>Piktogramme Arbeitsaufträge</i>
Auge	openclipart.org-Nutzer secretlondon. <i>Blue eye</i> . 2009. URL: <a href="https://t1p.de/gxoy">https://t1p.de/gxoy</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Hand	openclipart.org-Nutzer mireille. <i>hand</i> . 2010. URL: <a href="https://t1p.de/v0pq">https://t1p.de/v0pq</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz. Anpassung durch Fricke.
Glühbirne	openclipart.org-Nutzer palomaironique. <i>Light Bulb On</i> . 2010. URL: <a href="https://t1p.de/vipe">https://t1p.de/vipe</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Mund	openclipart.org-Nutzer xeolhades. <i>mouth</i> . 2008. URL: <a href="https://t1p.de/ktmn">https://t1p.de/ktmn</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Schere	openclipart.org-Nutzer schoolfreeware. <i>Scissors</i> . 2008. URL: <a href="https://t1p.de/2hyu">https://t1p.de/2hyu</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Schloss offen/geschlossen	openclipart.org-Nutzer lyo. <i>Open and closed lock</i> . 2013. URL: <a href="https://t1p.de/z6sk">https://t1p.de/z6sk</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz. Anpassung durch Fricke.
Sozialformen: EA, PA, GA, Plenum	openclipart.org-Nutzer Thangalin. <i>Happy Face</i> . 2016. URL: <a href="https://t1p.de/q4z3">https://t1p.de/q4z3</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.
Stift	openclipart.org-Nutzer hypocore. <i>Pencil</i> . 2015. URL: <a href="https://t1p.de/pdpu">https://t1p.de/pdpu</a> (besucht am 09. 03. 2019). Abbildung steht unter CC0 Lizenz.

## Literatur zum Modul Kryptologie

- Albers, Stine, Hrsg. *Sachunterricht studieren: Bildung im Blickpunkt – das Sokratische Gespräch*. 2013. URL: <https://t1p.de/7n59> (besucht am 09. 03. 2019).
- Bischof, Ernestine und Roland Mittermeir. *Informatik erLeben. Beispiele für schülerinnen- und schüleraktivierenden Informatikunterricht*. 2010. URL: <https://t1p.de/5gzo> (besucht am 09. 03. 2019).
- Borys, Thomas. *Codierung und Kryptologie. Facetten einer anwendungsorientierten Mathematik im Bildungsprozess*. 1. Aufl. Wiesbaden: Viewg+Teubner Research, 2011. ISBN: 978-3-8348-1706-8.
- Calmet, Claire, Mathieu Hirtzig und David Wilgenbus. 1, 2, 3 ... *kodiert, programmiert, verschlüsselt! – Informatik in der Grundschule*. 17. Jan. 2017. URL: <https://t1p.de/1v2h> (besucht am 09. 03. 2019).
- Denton Gillian und Hallinan, Camille. *Memo – Wissen: Spione*. Dorling Kindersley Verlag, 2011. ISBN: 3831019061.
- Didaktik der Informatik an der Bergischen Universität Wuppertal, Hrsg. *Internetpräsenz der DdI für den Primarbereich*. 2018. URL: <https://uni-w.de/zn> (besucht am 09. 03. 2019).
- Gallenbacher, Jens. *Ausstellung: Abenteuer Informatik – Informatik begreifen*. Darmstadt, 2008. URL: <https://t1p.de/yqik> (besucht am 09. 03. 2019).
- Gesellschaft für Informatik e. V., Hrsg. *Kompetenzen für informatische Bildung im Primarbereich. Empfehlungen der Gesellschaft für Informatik e. V. Erarbeitet vom Arbeitskreis »Bildungsstandards Primarbereich« – Beschluss des GI-Präsidiums vom 31. Januar 2019 – wird in gedruckter Form der LOG IN 39 (2019) Heft 191/192 beigelegt. 7. Feb. 2019. URL: <http://uni-w.de/1gm> (besucht am 09. 03. 2019).*
- Hielscher, Michael. *Mal-den-Code.de*. 9. Mai 2015. URL: <https://t1p.de/vwzy> (besucht am 09. 03. 2019).
- Humbert, Ludger. *Didaktik der Informatik – mit praxiserprobtem Unterrichtsmaterial*. 2., überarbeitete und erweiterte Aufl. Leitfäden der Informatik. Wiesbaden: B.G. Teubner Verlag, Aug. 2006. ISBN: 3-8351-0112-9.
- *Veranstaltungskarte – Informatik im Alltag – Wintersemester 2018/2019 – Bergische Universität Wuppertal*. Okt. 2018. URL: <http://uni-w.de/1ch> (besucht am 09. 03. 2019).
- Kippenhahn, Rudolf. *Streng geheim! Wie man Botschaften verschlüsselt und Zahlencodes knackt*. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag, 2003. ISBN: 3499211645.
- Kohlhammer, Michael. *Geolino: Spionage: Knifflige Codes und geheime Fälle*. Franckh Kosmos Verlag, 2011. ISBN: 978-3440128244.
- Luringen, Wikipedia-Nutzer. *Skytale.png*. 2007. URL: <https://t1p.de/i3t6> (besucht am 09. 03. 2019).
- Müller, Dorothee. *SpionCamp-Stationen*. Juli 2012. URL: <https://uni-w.de/1u> (besucht am 09. 03. 2019).
- Schlitzer, Monika. *Memo – Wissen Entdecken: Computer*. Dorling Kindersley Verlag, 2012. ISBN: 3831020477.
- Schüller, Julia. »Informatiktricks – phänomenorientierter Informatikunterricht zu Beginn der Sekundarstufe I«. Master-Thesis. Wuppertal: Fachgebiet Didaktik der Informatik – Bergische Universität, Aug. 2014. URL: <https://t1p.de/wwpn> (besucht am 09. 03. 2019).
- Singh, Simon. *Codes. Die Kunst der Verschlüsselung*. 2014. ISBN: 9783423621670.
- Strecker, Kerstin. *Kryptografie. Virtuelle Lehrerweiterbildung Informatik in Niedersachsen*, 2009. URL: <https://t1p.de/p3sp> (besucht am 09. 03. 2019).
- Verg, Martin. *GEOLINO extra 34-2012 – Spione und Detektive*. Gruner + Jahr, 2012. ISBN: 978-3652001199.
- Yurdakul Cakir-Dikkaya u. a. *DaZ für den Fachunterricht der Sekundarstufe I: Gesellschaftswissenschaften Klasse 5-10: Materialien, Tipps, Hilfen und Methoden. Kopiervorlagen mit CD-ROM*. Cornelsen Scriptor, 2017. ISBN: 358915215X.

### Lizenzhinweis

Die Lehrerhandreichung Kryptologie steht unter der folgenden Creative-Commons-Lizenz: . Unter <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.de> sind Details zu der Lizenz dokumentiert.



# Informatik an Grundschulen (IaG) Hinweise zur Gestaltung von Fortbildungen und Schulungen für Lehrkräfte



Rheinisch-Westfälische Technische Hochschule Aachen  
Universität Paderborn  
Bergische Universität Wuppertal

Ministerium für  
Schule und Bildung  
des Landes Nordrhein-Westfalen



# Inhaltsverzeichnis

<b>Vorwort – Gestaltung von Fortbildungen und Schulungen für Lehrkräfte</b>	<b>FB 02</b>
<b>Teil A: Projekt- und modulspezifische Aspekte der Gestaltung der Lehrkräfte-Schulung</b>	<b>FB 02</b>
Allgemeine Zielsetzungen der Schulung . . . . .	FB 02
Genese des Schulungskonzepts und der Schulungsmaterialien . . . . .	FB 02
Organisatorische Rahmenbedingungen der Schulungen im Projekt . . . . .	FB 02
Modulspezifische Umsetzungen der Schulungen . . . . .	FB 03
Fachliche Vertiefung, Vernetzung und unterrichtliche Fortführung der Themen . . . . .	FB 03
<b>Teil B: Grundlegende, modulübergreifende Aspekte der Gestaltung von Schulungsveranstaltungen</b>	<b>FB 04</b>
Warm-Up: Ankommen . . . . .	FB 04
Organisatorisches . . . . .	FB 04
Aktivierung . . . . .	FB 04
Input und Hands-on . . . . .	FB 04
Integration des Lern- und Erfahrungszuwachses . . . . .	FB 04
Vorbereitung und Planung der eigenen Umsetzung im Unterricht . . . . .	FB 05
Bei Schulungsreihen: Vereinbarungen zur Weiterarbeit . . . . .	FB 05
Cooldown: Abgleich, Ausblick . . . . .	FB 05

## Vorwort – Gestaltung von Fortbildungen und Schulungen für Lehrkräfte

Dieses Dokument zur Gestaltung von Fortbildungen und Schulungen für Lehrkräfte zu ausgewählten Elementen des Fachs Informatik wurde als ein Ergebnis im Rahmen des Projekts Informatik an Grundschulen (IaG) des Ministeriums für Schule und Bildung des Landes Nordrhein-Westfalen zusammengestellt. Im Rahmen des Projekts wurden Schulungen durchgeführt, die hier zu Hinweisen zusammengestellt werden, die auch für Fortbildungen zu Informatik an Grundschulen herangezogen werden können. Anmerkungen und Hinweise in diesem Teil der Materialien beziehen sich auf die drei Module, für die auch die Unterrichtsmaterialien zur Informatik an Grundschulen im Projekt IaG entwickelt wurden.

Diese Hinweise beinhalten sowohl projekt- und modulspezifische Aspekte als auch grundlegende, modulübergreifende Hinweise zur organisatorischen und didaktisch-methodischen Gestaltung von Schulungsveranstaltungen.

## Teil A: Projekt- und modulspezifische Aspekte der Gestaltung der Lehrkräfte-Schulung

### Allgemeine Zielsetzungen der Schulung

Die Schulung der Lehrkräfte orientiert sich an folgenden Zielsetzungen:

- Fachliche Auseinandersetzung mit den informatischen Inhalten des jeweiligen Moduls und mit diesen in engem Zusammenhang stehenden übergreifenden Prinzipien der Informatik (wie z.B. Algorithmik, EVA-Prinzip, Informatiksysteme, etc.)
- Inhaltliche und methodische Vorbereitung der Lehrkräfte (LK) auf die eigenständige Durchführung des Moduls mit den Schülerinnen und Schülern der eigenen Lerngruppe
- Austausch und Reflexion über die Erprobung und ggf. vorgenommene lerngruppenspezifische Anpassung des Unterrichtsmaterials
- Ggf. gemeinsame Erarbeitung unterrichtspraktischer Umsetzungsvarianten

### Genese des Schulungskonzepts und der Schulungsmaterialien

Das Schulungskonzept wurde im Projekt IaG in Kooperation mit den Lehrkräften an den drei Projektstandorten (Aachen, Paderborn, Wuppertal) entwickelt, iterativ erprobt und verbessert. Es lehnt sich inhaltlich an die Lernmaterialien der drei Module für Schülerinnen und Schüler an und bezieht sich daher auf die Themen »Digitale Welt«, »Roboter« und »Kryptologie«. Es ist auf fachlicher Ebene, wie die Materialien und Konzepte der Module für Schülerinnen und Schüler auch, primär am didaktisch-methodischen Konzept von »Computer Science unplugged« orientiert.

### Organisatorische Rahmenbedingungen der Schulungen im Projekt

Die Schulungen wurden in enger Abstimmung standortspezifisch für die jeweiligen Module entwickelt, erprobt und nach wechselseitigen Rückmeldungen modifiziert. Obwohl es aus organisatorischen Gründen z. T. leichte Variationen in der Umsetzung der jeweiligen Schulungen zu den Modulen gab, kristallisierte sich folgendes gemeinsames organisatorisches Schema als Grundkonzept für die Schulungen zu den drei Themenbereichen heraus:

- ▶ je Modul ein Tagesseminar zur Einführung in das Thema (Dauer: ca. 6 Stunden)

- ▶ Phase der praktischen Erprobung des Moduls in den eigenen Lerngruppen durch die Lehrkräfte (Dauer: mehrere Wochen)
- ▶ Halbtagsseminar zum Austausch und Reflexion über Unterrichtserfahrungen, Diskussion möglicher Umsetzungsvarianten und unterrichtliche Weiterführung des Themas; Fachliche Vertiefung des Themas (Dauer: 3 Stunden)

Die vorgeschlagene zeitliche Strukturierung der Schulung hat sich im Projekt IaG bewährt. Wenn aufgrund organisatorischer Erfordernisse eine andere zeitliche Blockstruktur erforderlich sein sollte, wären aber auch andere Organisationsformen, z. B. drei Halbtage mit je 3 Stunden, durchaus realisierbar.

## Modulspezifische Umsetzungen der Schulungen

Details zur modulspezifischen Planung und Durchführung von Schulungsveranstaltungen sind jeweils in tabellarischer Form dargestellt. Zur Vorbereitung der Schulung sollten für die Lehrkräfte jeweils die modulbezogenen Lernmaterialien in ausreichendem Umfang zur Verfügung stehen, die später auch im Unterricht mit den Schülerinnen und Schüler zum Einsatz kommen. Die modulbezogenen didaktischen Handreichungen sollten spätestens zu Beginn der jeweiligen themenbezogenen Schulung verteilt werden. Es erscheint durchaus auch sinnvoll, den Lehrkräften die Schulungsmaterialien zu den jeweiligen Modulen, soweit es sich nicht um »Hardware«, wie Bastelmaterial, handelt, einige Zeit vor Beginn der Schulung zur Verfügung zu stellen. Sie haben dann Gelegenheit, sich bereits im Vorfeld der Schulung mit den informatischen und unterrichtsmethodischen Aspekten des Themas vertraut zu machen und entstandene eigene Fragen zur Thematik in die Schulung einzubringen.

## Fachliche Vertiefung, Vernetzung und unterrichtliche Fortführung der Themen

Das Schulungskonzept zielt besonders im letzten Abschnitt auf die fachliche Vertiefung und Vernetzung der jeweiligen Informatikinhalte ab. Die begleitende Evaluation im Projekt IaG hat zudem ergeben, dass sowohl auf Unterrichtsebene als auch in den Schulungen der Lehrkräfte eine Kontextualisierung und Konkretisierung der angesprochenen informatischen Aspekte anhand weiterer konkreter Informatiksysteme aus der Erfahrungswelt der Schülerinnen und Schüler bzw. der Lehrkräfte erfolgen sollte. Es sollte dabei besonderes Augenmerk auf den Anwendungsbezug und die Bedeutung der jeweiligen informatischen Konzepte im Alltag gelegt werden. Hier dazu einige Beispiele:

**Modul »Digitale Welt«** – Erkundung des Informatiksystems *Ampel* (Codierung, Zustände, Automat)

**Modul »Roboter«** – Programmierung mit *ScratchJr* oder von *Ozobots* (Algorithmik, Entwicklungsumgebung, Visualisierung)

**Modul »Kryptologie«** – Verschlüsselung in der Praxis (Ver- und Entschlüsseln von E-Mails, Digitale Unterschriften, sichere Datenübertragung im World Wide Web)

Damit sich die Lehrkräfte in der Schulung inhaltliche, themenübergreifende Zusammenhänge zu erschließen können, sollten wichtige themenübergreifende informatische Konzepte wie Algorithmik, EVA-Prinzip oder Informatiksysteme, inhaltlich vernetzt, und im Kontext aller drei Module besprochen werden.

Die unterrichtliche Fortführung der Themen ermöglicht auch einen Übergang von der *CS Unplugged*-Methode zur Arbeit mit Informatiksystemen.

Die Projektevaluation zeigt, dass die Schülerinnen und Schüler den Informatikunterricht mit der Arbeit am Computer verbinden, sie selbst im Unterricht solche Arbeitsphasen wünschen und diese für sie ein hohes Motivationspotenzial besitzen.

Daher sollten auch in der Lehrkräfteschulung zu den drei Modulen im jeweils letzten Zeitblock solche Arbeitsformen zumindest thematisiert, besser noch auch praktisch erprobt werden.

## Teil B – Grundlegende, modulübergreifende Aspekte der Gestaltung von Schulungsveranstaltungen

### Warm-up: Ankommen

Häufig kommen die Teilnehmerinnen und Teilnehmer aus einem stressigen Alltag zu Ihnen. Bereiten Sie eine lernförderliche, kommunikative Umgebung für die Teilnehmerinnen und Teilnehmer vor. Sorgen Sie für Getränke und Kekse, schaffen Sie Orientierung in den Räumlichkeiten. Führen Sie vor dem offiziellen Beginn Smalltalk, um das Eis zu brechen. Es bieten sich hierfür auch kleine Warm-up Spiele an, die nicht zwingend bereits den Inhalt der Veranstaltung aufgreifen.

### Organisatorisches: Bedürfnisse klären

Schaffen Sie inhaltliche und organisatorische Transparenz. Klären Sie Fragen wie die Parksituation, Bescheinigungen und Teilnehmerlisten vorab, so dass die Teilnehmerinnen und Teilnehmer sich voll auf den Inhalt einlassen können. Geben Sie transparent die geplanten Pausen an oder planen Sie gemeinsam mit den Teilnehmerinnen und Teilnehmer den Tagesablauf.

### Aktivierung: Beziehung zum Inhalt herstellen

Stellen Sie eine Verbindung zwischen Teilnehmerinnen und Teilnehmer und der Sache her. Nutzen Sie dafür vielfältige Medien, wie Zeitungartikel, Comics, kurze Filme, Statements. Gerne dürfen diese auch provokante Aussagen enthalten, an denen die Teilnehmerinnen und Teilnehmer sich reiben müssen/dürfen. Lassen Sie möglichst viele Teilnehmerinnen und Teilnehmer zu Wort kommen und deren Einstellung zur Sache äußern. Nehmen Sie die Äußerungen dankend auf, ohne darauf einzugehen. Notieren sie diese ggf. auf einer Flipchart.

Bei Schulungsreihen: Greifen Sie an dieser Stelle die Erfahrungen der Teilnehmerinnen und Teilnehmer aus vorangegangenen Terminen auf und schaffen Sie einen informierenden Rückblick.

Bei Evaluationsvorhaben: Nutzen Sie diesen Slot zum Einholen von Erfahrungsberichten: »Wie haben Anpassungen und Durchführungen funktioniert?«

Geben Sie Raum für nachträglich aufgekommene Unklarheiten oder Rückfragen. »Welche Fragestellungen sind im Nachhinein entstanden?«

### Input/Hands-on oder Hands-on/Input

Lassen Sie die Teilnehmerinnen und Teilnehmer die Sache *be-greifen*. Lassen Sie sie dazu am Material arbeiten – explorativ oder bereits mit einer Fragestellung. Unterfüttern Sie diese Phase mit Sachinput. Dies kann auf mehrere Arten passieren.

*Induktiv:* Beginnen Sie mit einem fachlichen Input als Vortrag. Ergänzende Information kann über eine Wandzeitung zur eigenen Recherche zur Verfügung gestellt werden. Lassen Sie die Teilnehmerinnen und Teilnehmer mit dem fachlichen Wissen das Material erproben und bewerten.

*Deduktiv:* Schicken Sie die Teilnehmerinnen und Teilnehmer an das Material, lassen Sie sie explorieren und ausprobieren, die Wandzeitung dient als erste Ergänzung. Klären Sie dann in einer Plenumsphase Fragestellungen zum Material. Anschließend nutzen Sie den Fachvortrag zur Anreicherung des Erfahrenen.

Richten Sie den Blick der Teilnehmerinnen und Teilnehmer auf die folgenden Bereiche:

*Selbsterfahrung, Sachebene, Didaktik, Methodik*

## Integration des Lern- und Erfahrungszuwachses

Schließen Sie nun eine Plenumsdiskussion über das Material an. Lehrerinnen und Lehrer neigen häufig dazu, Unterrichtsmaterial für den Einsatz im Unterricht zu bewerten, bevor sie selbst die Sachebene durchdrungen haben. Verweilen Sie daher mit den Teilnehmerinnen und Teilnehmer zunächst auf der Ebene der Selbsterfahrung. Nutzen Sie diese Erfahrungen im Sinne eines methodischen Doppeldeckers: »Was hast *Du* gelernt?«, »Was macht das mit *Dir*?«, »Welche Schlüsse ziehst Du daraus für *Deine* Unterrichtsplanung?«, »Wie kannst Du diese Erfahrung in Deinen Unterricht einfließen lassen?«

Lenken Sie die Wahrnehmung ebenso auf die didaktischen Implikationen: Thematisieren Sie die fachdidaktischen Konzepte und daraus resultierende methodische Entscheidungen am Material und an der Selbsterfahrung. Integrieren Sie an dieser Stelle Querschnittsthemen wie z. B. Fachsprache oder Methodenlernen. Nach diesem intensiven Block bietet sich meist eine kleine Verschnaufpause an.

## Vorbereitung und Planung der eigenen Umsetzung im Unterricht

Nutzen Sie die freigelegten Hürden und Stolpersteine aus der vorherigen Phase für die nun folgende Phase. Lassen Sie die Teilnehmerinnen und Teilnehmer in Partner- oder Kleingruppen Vorbereitungen für die Durchführung der Modulinhalte in ihren eigenen Lerngruppen treffen. Lassen Sie optimalerweise Teilnehmerinnen und Teilnehmer von gleichen Schulen gemeinsam in die Planung gehen. Geben Sie den Teilnehmerinnen und Teilnehmer ausreichend Zeit für ihre Planungen.

Richten Sie den Blick auf die folgenden Kernfragen:

- ▶ »Welche Hürden erwarte ich bei meinen Schülerinnen und Schülern?«
- ▶ »Welche Vorbereitungen muss ich treffen?«
- ▶ »Welche Anpassungen muss ich vornehmen?«
- ▶ »Welche Form von Unterstützung benötige ich und wo bekomme ich sie?«

Lassen Sie die Ergebnisse kurz vorstellen und besprechen. Möglicherweise ergeben sich Synergieeffekte zwischen den Kleingruppen. Außerdem sorgt der Abgleich der eigenen Arbeitsstände zu einem stärkerem Gemeinschaftsgefühl.

## Bei Schulungsreihen: Vereinbarungen zur Weiterarbeit

Im Rahmen von Schulungsreihen empfiehlt es sich Absprachen für die Weiterarbeit in den nächsten Terminen zu treffen.

Besprechen Sie, in welcher Form die Durchführungen dokumentiert werden sollen, um für die nächste Sitzung abbildbar zu werden. Schaffen Sie ebenso Transparenz für den Einsatz von Schüler/Lehrer-Evaluationsbögen. Lagern Sie ggf. notwendige Zusatzinformation und Verweise in nachbereitende E-Mails aus.

## Cooldown: Abgleich, Ausblick

Sprechen Sie in einem abschließendem Blitzlicht o. Ä. die Gefühlsebene der Teilnehmerinnen und Teilnehmer an. Geben Sie einen organisatorischen Ausblick auf die weitere Struktur Ihres Schulungsangebots. Erinnern Sie an Kontaktmöglichkeiten zwischen zwei Schulungsterminen.