

pädagogische hochschule schwyz

Informatik ohne Strom



Michael Hielscher
Beat Döbeli Honegger

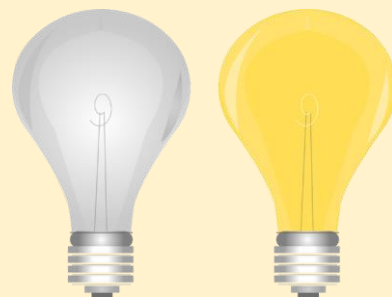
05. Mai 2018



Worum geht es?

Computer kennen nur Nullen und Einsen. Wie kann der Computer aber dann zählen oder etwas ausrechnen?

In dieser Übung lernst du, mit den Fingern nicht nur bis 10, sondern bis 1023 zu zählen. Danach weisst du auch, was das Binärsystem ist.



Wie funktioniert es?

1. Zählen im Binärsystem

Unser dezimales Zahlensystem kennt 10 verschiedene Ziffern (0 bis 9). Wenn wir eine Zahl grösser als 9 aufschreiben wollen, benötigen wir mehr als eine Ziffer – also 10, 11, 12, 13 ... Das Binärsystem kennt hingegen nur die beiden Ziffern **0** und **1**. Trotzdem kann man auch im Binärsystem beliebig grosse Zahlen darstellen und für jede Dezimalzahl gibt es eine entsprechende Binärdarstellung. Da wir binär nur **0** und **1** haben, müssen wir bereits für eine Zwei eine weitere Stelle verwenden. Für die dezimale 2 schreiben wir binär **10**, für 3 ergibt sich **11** und für 4 müssen wir bereits eine dritte Stelle mit **100** notieren. Binärzahlen werden deshalb schnell sehr lang. Das geht nun immer so weiter $5 = 101$, $6 = 110$, $7 = 111$, $8 = 1000$. Kannst du sagen, wie die Binärzahl für 13 aussieht?

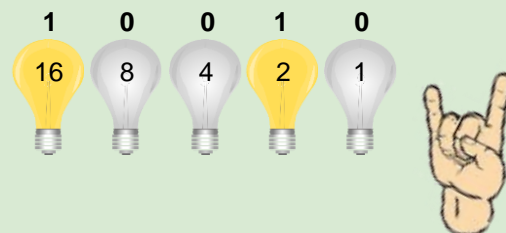
2. Binäre Zahlen im Computer

Computer arbeitet nur mit Folgen von Nullen und Einsen – entweder der Strom fliesst (**1**) oder eben nicht (**0**).

Eine Binärzahl wie **10010** kann man sich als 5 Lampen vorstellen:

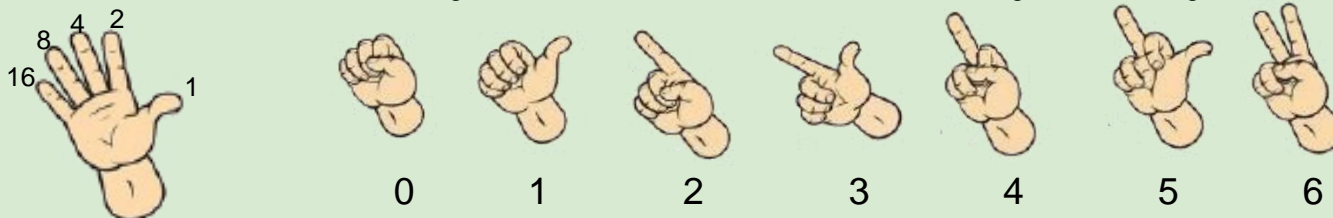
Überall wo eine 1 steht, brennt die Lampe. Auf jeder Lampe steht eine Zahl, die den Wert der Lampe angibt. Dabei steht die kleinste Zahl ganz rechts und eine Lampe links davon hat jeweils den doppelten Wert.

Um herauszufinden, welche Zahl **10010** darstellt, musst du nur die Zahlen aller brennenden Lampen zusammenzählen. Im Beispiel $16+2 = 18$.

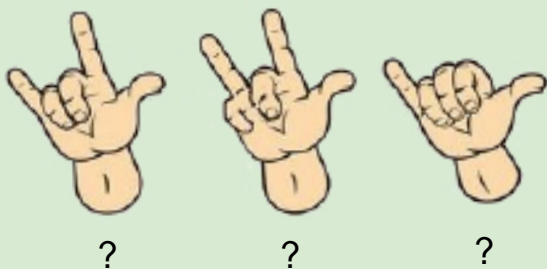


3. Mit den Fingern binär zählen

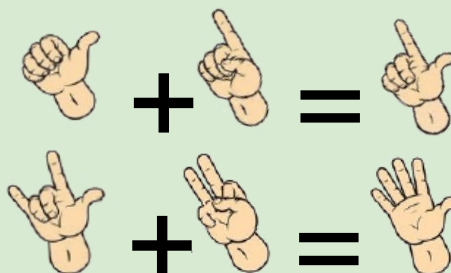
Statt Lampen können wir auch unsere Finger verwenden. Ein ausgestreckter Finger ist eine **1** und ein eingezogener eine **0**. Zum Üben kannst du dir mit einem Stift die Zahlen 1, 2, 4, 8 und 16 wie in folgendem Bild an die Fingerspitzen schreiben. Versuche nun mit deinen Fingern binär bis 31 zu zählen. Das erfordert anfangs etwas Übung.



Welche Zahlen zeigen die folgenden Hände?
(addiere die Werte an jedem ausgestreckten Finger)



Mit Binärzahlen kann man sogar rechnen.
Verstehst du die folgenden Abbildungen?



Sobald du mit einer Hand binär zählen kannst, verwende auch die zweite Hand und setze die Zahlenfolge mit 32, 64, 128, 256 und 512 fort. So kannst du mit beiden Händen Zahlen bis 1023 darstellen.

Worum geht es?

Du hast bereits Binärzahlen kennengelernt. Mit diesem Wissen kannst du einen scheinbar verblüffenden Zaubertrick aufführen, der für andere nur schwer zu durchschauen ist.



Wie funktioniert es?

Auf iLearnT.ch/stromlos gibt es eine zusätzliche Druckvorlage zu dieser Aufgabe.

1. Vorbereitung

Du benötigst für den Trick sechs besondere Zahlenkarten. Stelle dir einen eigenen Satz dieser Karten auf Papier her. Schau dir die Karten einmal genauer an, fällt dir etwas an den Zahlen auf? Gibt es ein System?

1	3	5	7	9	11	13	15
17	19	21	23	25	27	29	31
33	35	37	39	41	43	45	47
49	51	53	55	57	59	61	63

2	3	6	7	10	11	14	15
18	19	22	23	26	27	30	31
34	35	38	39	42	43	46	47
50	51	54	55	58	59	62	63

4	5	6	7	12	13	14	15
20	21	22	23	28	29	30	31
36	37	38	39	44	45	46	47
52	53	54	55	60	61	62	63

8	9	10	11	12	13	14	15
24	25	26	27	28	29	30	31
40	41	42	43	44	45	46	47
56	57	58	59	60	61	62	63

16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

2. Der Trick

Als Vorbereitung mische die 6 Karten zufällig und lege sie auf einen Stapel. Bitte jemanden aus dem Publikum sich eine beliebige Zahl zwischen 0 und 63 auszudenken und nicht zu verraten. Sie oder er soll die Zahl verdeckt auf einen Zettel schreiben, ohne das du es sehen kannst. Jetzt zeigst du nacheinander die 6 Karten und fragst, ob die gedachte Zahl darauf zu sehen ist. Jedes Mal wenn der Kandidat JA sagt, legst du die Karte auf eine Seite, wenn er NEIN sagt auf die andere. Zähle schnell die jeweils erste Zahl der Karten auf dem JA Stapel zusammen. Beispiel: Der Kandidat sagt bei der Karte mit der 1, der 4 und der 16 JA, bei den anderen NEIN. Du addierst also $1+4+16 = 21$. Die gedachte Zahl ist die 21. Verkünde dem Publikum, an welche Zahl der Kandidat gedacht hat und lass ihn anschliessend den Zettel zeigen, auf dem er die Zahl am Anfang notiert hat. Stimmt deine Zahl nicht, hast du dich entweder verrechnet, oder der Kandidat hat bei einer Karte die Frage falsch beantwortet. Da du zwei Stapel abgelegt hast, kannst du selbst noch einmal überprüfen, bei welcher Karte der Kandidat seine Zahl eventuell übersehen hat und ihn darauf aufmerksam machen, dass der Fehler nicht bei dir lag ☺ Wiederhole den Trick ruhig ein paar mal und lass die Kandidaten rätseln, wie der Trick funktioniert. An welche Zahl hat der Kandidat in folgendem Beispiel gedacht?

JA

4	5	6	7	12	13	14	15
8	9	10	11	16	17	18	19
20	21	22	23	24	25	26	27
28	29	30	31	32	33	34	35
36	37	38	39	40	41	42	43
44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59
60	61	62	63				

NEIN

2	3	6	7	10	11	14	15
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	

1 0 1 1 0 0

32 16 8 4 2 1

3. Hintergrund

Aber wie funktioniert der Trick nun genau? Würdest du auf die 6 Zahlenkarten Binärzahlen statt Dezimalzahlen schreiben, dann würde dir sehr schnell etwas an den Zahlen auffallen. Auf der ersten Karte sind alle Zahlen notiert, die in der Binärdarstellung in der ersten Stelle von rechts eine 1 stehen haben. Auf der zweiten Karte alle, die an der zweiten Stelle eine 1 stehen haben usw. Die 63 ist binär die **111111** da hier alle Stellen eine 1 haben, ist diese Zahl auf jeder der sechs Karten zu sehen. Die 0 ist binär eine **000000** und steht damit auf keiner Karte. Die 0 wäre deshalb die richtige Lösung, wenn der Kandidat bei allen 6 Karten NEIN sagt. Wie könnte man den Trick mit grösseren Zahlen als der 63 durchführen?

Worum geht es?

Informationen wurden schon lange vor dem Computer „codiert“: Einem Symbol, einem Zeichen oder einem Laut wurde eine ganz bestimmte Bedeutung zugeordnet (z.B. Hieroglyphen, Musiknoten, Sprache).

Da der Computer nur genau zwei Symbole - **0** und **1** - kennt, müssen alle Informationen aus eindeutigen Kombinationen von mehreren 0en und 1en aufgeschrieben, also „codiert“ werden. „Codierung“ kann man auch für einfache Geheimschriften verwenden. Nur wer die richtige Zuordnung von Zeichen zu Code kennt, kann sie entziffern.



Wie funktioniert es?

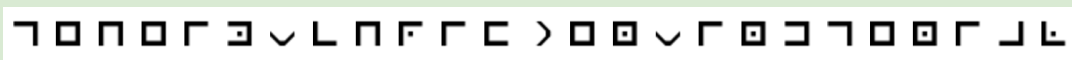
1. Die Geheimschrift des Computer

Hast du schon einmal eine geheime Botschaft verschickt? Vielleicht hast du schon einmal einen Brief in Spiegelschrift geschrieben oder einen, in dem alle Anfangsbuchstaben der Wörter einen geheimen Satz gebildet haben. Es gibt ganz viele Möglichkeiten, etwas so aufzuschreiben, dass zwar alle deine Botschaft ansehen, aber nur gewisse Empfänger sie verstehen können. Der Computer verwendet auch eine Art Geheimschrift, die für uns unverständlich klingt: **01000001 01000010 01000011** diese drei Zahlenfolgen stehen für „ABC“ im sogenannten ASCII-Code. In einer Tabelle wird jedem Zeichen (Buchstabe, Satzzeichen, Ziffer usw.) genau eine bestimmte Zahlenfolge aus 0 und 1 zugeordnet. Wir müssen also nur in die Tabelle schauen und ablesen, welches Zeichen sich hinter der Zahl versteckt. Kannst du den Code für X,Y und Z in der Tabelle rechts fortsetzen und ergänzen? Kannst du deinen Vornamen mit Hilfe der Tabelle als lange Folge von 0en und 1en aufschreiben? Der ASCII-Code ist schon über 50 Jahre alt und stammt aus den Anfangszeiten der Computer. Er wird aber auch heute noch an vielen Stellen verwendet. Welche Zahlenfolge damals welchem Zeichen zugeordnet wurde, ist eigentlich egal. Wichtig ist nur, dass die Zuordnung eindeutig ist (d.h. jedes Zeichen hat einen eigenen Code) und dass alle die gleiche Codetabelle verwenden.

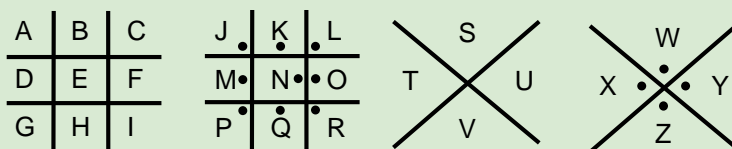
ASCII-Code	Zeichen
00100000	Leerzeichen
01000001	A
01000010	B
01000011	C
01000100	D
01000101	E
01000110	F
01000111	G
01001000	H
01001001	I
01001010	J
01001011	K
01001100	L
01001101	M
01001110	N
01001111	O
01010000	P
01010001	Q
01010010	R
01010011	S
01010100	T
01010101	U
01010110	V
01010111	W
	X
	Y
	Z

2. Einfache Geheimschriften selbst erstellen

Kannst du das Prinzip des Codierens und Decodierens auch für Geheimschriften anwenden? Der Freimaurer-Code wurde im Mittelalter verwendet. Nur wer den Code kannte, konnte die geheime Botschaft lesen. Kannst du die folgende Botschaft entschlüsseln, wenn du die Code-Gitter darunter zur Hilfe nimmst? Kannst du eine eigene, ähnliche Geheimschrift erfinden?

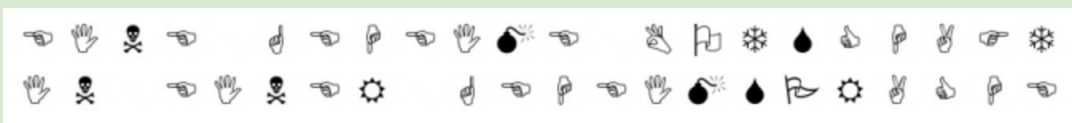


Freimaurer-Code:



3. Codes knacken

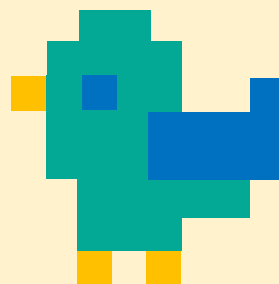
Einfache Geheimschriften, bei denen ein Zeichen jeweils genau einem anderen Zeichen oder Code zugeordnet wird, lassen sich relativ leicht durch Raten entziffern. In deutschen Texten kommt der Buchstabe „E“ am häufigsten vor. Ebenfalls häufig sind N, I, S, R, A und T. Schaffst du es, die folgende Botschaft zu entziffern?



Worum geht es?

Wie kann ein Computer farbige Bilder speichern, wenn der doch nur **0** und **1** kennt? Auf dieser Seite lernst du, wie das geht!

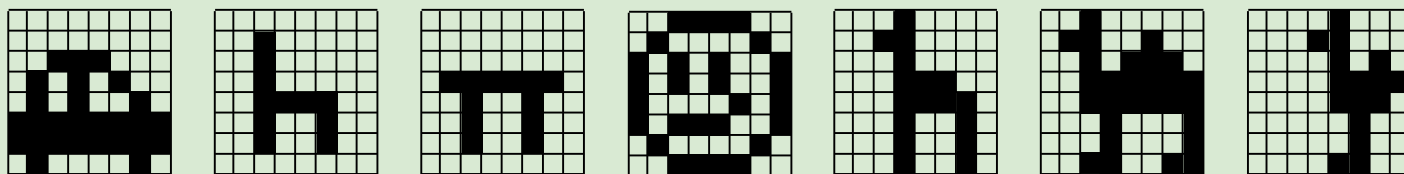
Bilder auf dem Bildschirm bestehen aus ganz vielen kleinen, quadratischen Kästchen – so genannte Pixel. Jedes Pixel hat immer genau eine Farbe. Da auf einen Bildschirm heute mehrere Millionen Pixel passen, sind sie nur noch mit der Lupe zu erkennen.



Wie funktioniert es?

1. Pixelbilder zeichnen

Nimm ein Blatt Häuschenpapier und einen Bleistift. Versuche einige kleine Bilder zu zeichnen, welche nur aus leeren und komplett ausgefüllten Kästchen besteht. Hier einige Beispiele:



Ein solches Schwarz/Weiss-Bild lässt sich für den Computer ganz einfach aufschreiben. Für jedes ausgefüllte Kästchen schreiben wir eine **1** und für jedes leere Kästchen eine **0**. Schau dir das folgende Beispiel mit dem Auto an.

Kannst du ein eigenes Pixelbild mit 8 x 8 Pixeln mit **0** und **1** aufschreiben?

	<pre> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 0 1 0 </pre>	<p>Zur besseren Lesbarkeit wurde die Zahlenfolge nach jeweils 8 Zahlen auf der nächsten Zeile fortgesetzt. Im Computer stehen sie aber alle direkt hintereinander. Spielt selber Computer und schickt euch gegenseitig Bilder: Schreibt ein 8 x 8 Pixel-Bild als eine lange Folge von 0 und 1 auf und tauscht diese untereinander aus. Zeichnet das zugehörige Bild und vergleicht anschliessend mit dem Original.</p>
--	--	--

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 0 0 1 0

2. Von Bits und Bytes

Eine einzelne **0** oder **1** nennt man Bit. Vielleicht hast du schon einmal von 32- und 64-Bit-Computern gehört. Damit ist gemeint, wie viele Bits der Prozessor gleichzeitig verarbeiten kann. Acht Bits hintereinander ergeben ein Byte. Das Byte ist in der Welt der Computer die am häufigsten verwendete Masseinheit. Hat man sehr viele davon, spricht man von Kilobyte, Megabyte, Gigabyte oder gar Terabyte. Ein Handyfoto benötigt mehrere Millionen Byte (etwa 3-5 Megabyte pro Bild). In der Photographie spricht man häufig auch von Megapixeln, was der Anzahl Pixel pro Foto bezeichnet.

3. Farbige Bilder nur mit 0 und 1 aufschreiben

Sind Pixelbilder farbig, so reicht eine 0 und eine 1 nicht mehr aus, um für ein Pixel die jeweilige Farbe zu notieren. Wir können statt einem Bit mehrere Bits pro Pixel verwenden und in einer Farbtabelle notieren, welche Bitfolge zu welcher Farbe gehören soll. Hier ein Beispiel für ein Bild mit 4 Farben. Kannst du zur rechten Binärfolge das passende Bild in das Feld zeichnen?

	<pre> 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 1 1 1 0 0 0 0 1 0 0 0 1 1 1 1 1 1 1 1 1 0 1 0 1 0 1 0 0 0 1 1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 </pre>	<table border="0"> <tr><td><input type="checkbox"/></td><td>0 0</td></tr> <tr><td><input type="checkbox"/></td><td>0 1</td></tr> <tr><td><input type="checkbox"/></td><td>1 0</td></tr> <tr><td><input type="checkbox"/></td><td>1 1</td></tr> </table>	<input type="checkbox"/>	0 0	<input type="checkbox"/>	0 1	<input type="checkbox"/>	1 0	<input type="checkbox"/>	1 1	<pre> 0 1 1 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 </pre>	
<input type="checkbox"/>	0 0											
<input type="checkbox"/>	0 1											
<input type="checkbox"/>	1 0											
<input type="checkbox"/>	1 1											

Für echte Fotos reichen 4 verschiedene Farben natürlich nicht aus. Ein einzelnes Pixel wird deshalb in den meisten Bildformaten mit 24 Bits (8 Bit pro Farbanteil **R**ot, **G**rün, **B**lau) gespeichert. Ein einziges rotes Pixel wird damit als **11111111 00000000 00000000** gespeichert – rot ist komplett vorhanden, alle anderen Farben sind leer. Durch Mischen entstehen insgesamt 16 Millionen verschiedene Farben. Ein türkises Pixel des Vogels oben hätte zum Beispiel: **00011011 10101011 10010101** (wenig Rot und viel Grün und viel Blau).

Worum geht es?

Im letzten Beispiel (Blatt 4) habt ihr gesehen, wie der Computer mit nur 0 und 1 Bilder speichern und verarbeiten kann. Doch wie geht das ganz konkret?

In diesem Beispiel sollt ihr ein eigenes Pixelbild aus kleinen Klebezetteln erstellen und anschliessend „einscannen“ und somit in 0en und 1en umwandeln. Mit Hilfe eines Scratch-Programms werdet ihr das so digitalisierte Bild anschliessend wieder am Bildschirm darstellen können.



Pixelbild aus Haftnotizen am Fenster

Wie funktioniert es?

1. Haftnotizen-Fensterbilder erstellen

Bildet 2er oder 3er Gruppen. Zeichnet ein kleines Pixelbild auf Papier (z.B. 8x8 oder 9x10, aber nicht mehr als 100 Pixel) mit genau 4 verschiedenen Farben. Nun sollt ihr dieses Bild möglichst schnell mit Haftnotizen an ein Fenster kleben. Teilt die Arbeit auf - eine Person klebt die Notizzettel, eine andere sagt, welche Farbe als nächstes benötigt wird.

Hinweis 1: Halbe Zettel, gedrehte Zettel, bemalte Zettel usw. sind nicht zulässig. Ein Pixel lässt sich nicht teilen!

Hinweis 2: Solltet ihr beim Ankleben Ansagen wie „3x rot und 8x gelb“ verwendet haben, dann habt ihr bereits ein einfaches Bildkompressionsverfahren (Laufängerkodierung) angewendet.

Euer Pixelbild am Fenster sieht vermutlich toll aus! Es wird aber nicht ewig dort hängen können. Wäre es nicht möglich, es zu „speichern“?

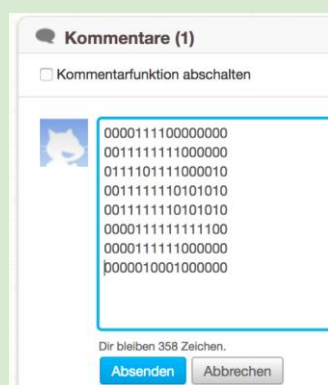
2. Haftnotizen-Fensterbilder „einscannen“ und speichern

Erstellt dazu zunächst eine Farbtabelle für eure Bildfarben und ordnet jeweils jeder Farbe eine eindeutige Bitfolge aus 0en und 1en zu. Zum Beispiel: 0 0 0 1 1 0 1 1



Um euer Fensterbild zu übertragen, öffnet zunächst in einem Webbrowser die Seite iLearnIT.ch/pixel

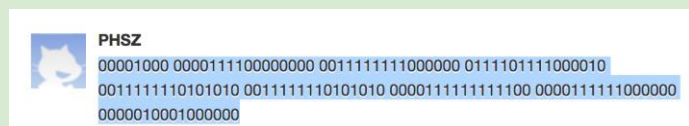
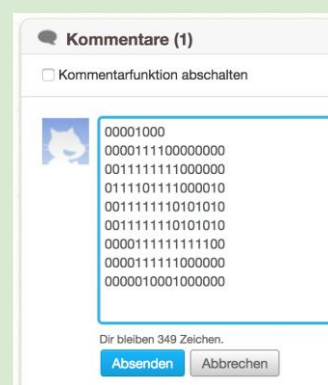
Jemand sollte nun die 0en und 1en Folge ins Kommentarfeld eintippen. Eine andere Person sagt die Folge ausgehend vom Fensterbild an. Wenn ihr zu dritt seid, kann jemand kontrollieren, dass keine Fehler bei der Übertragung passieren. Beginnt beim Eingeben nach jeder Pixelzeile ebenfalls eine neue Textzeile, jede Textzeile muss dann gleich lang sein (Tipp: zieht das Eingabefeld etwas grösser).



Damit das Scratch-Programm weiss, wie viele Pixel pro Zeile das Bild gross (breit) sein wird, benötigt es eine zusätzliche Information. Das Programm erwartet deshalb die Bildbreite als Binärzahl mit genau 8 Stellen.

Schreibe deine Bildbreite als Binärzahl auf und schreibe danach genügend 0en davor, um genau 8 Stellen zu erhalten. Beispiel für ein 9x7 Bild: 9 Pixel breit = 1001 = 00001001

Fügt die Binärzahl für die Bildbreite vor den Bilddaten ein (Beispiel rechts). Wenn ihr auf der Scratch-Website seid, könnt ihr die Bitfolge auch als Kommentar absenden. Alle Ziffern erscheinen dann hintereinander.



Startet das Scratch-Programm über das grüne Fähnchen. Bei der Eingabeaufforderung kopiert ihr per Copy&Paste eure ganze Bitfolge ins Dialogfeld (eventuell Ctrl+C und Ctrl+V verwenden, wenn Rechtsklick nicht funktioniert). Lasst euer Bild von Scratch zeichnen. Wenn alles stimmt, könnt ihr euer Fensterbild abhängen, ihr habt ja nun eine digitale Version gespeichert 😊



Zusatzfragen: Was ist eigentlich die maximale Bildbreite, die sich mit 8 Stellen aufschreiben lässt? Habt ihr eine Idee, warum die Bildbreite mit einer festgelegten Anzahl Stellen beschrieben wurde?

Worum geht es?

Du lernst einen verblüffenden Zaubertrick kennen, der nur 25 Karten (z.B. Spielkarten oder Postkarten) benötigt.

Gleichzeitig lernst du das Prinzip kennen, wie Computer Fehler beim Eintippen, Einlesen oder Übertragen von Daten erkennen oder gar korrigieren können.

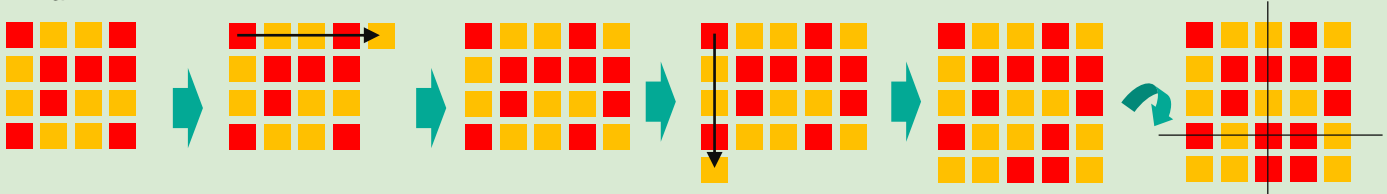


Wie funktioniert es?

1. Der Zaubertrick

Du benötigst 25 gleich grosse Karten mit deutlich unterscheidbaren Vorder- und Rückseiten (hell / dunkel). Lasse jemanden aus dem Publikum aus 16 Karten ein quadratisches Muster mit 4 x 4 Karten auflegen. Dabei sollen sich Vorder- und Rückseite möglichst zufällig abwechseln, so dass das entstehende Muster schwer zu merken ist.

Nun behauptest du, dass du die Sache noch etwas schwieriger machen möchtest und legst noch eine weitere Spalte rechts dazu. Du tust so, als seien diese zusätzlichen Karten ganz zufällig. In Wirklichkeit zählst du schnell in jeder Zeile die Anzahl Karten mit dunkler Seite. Ist die Anzahl gerade, legst du die zusätzliche Karte mit der hellen Seite daneben, sonst mit der dunklen Seite. (Würdest du nach dem Legen der fünften Karte nochmals zählen, musst du eine gerade Anzahl dunkler Karten haben). Das Gleiche machst du anschliessend mit den Spalten, um insgesamt 5 x 5 Karten aufzulegen. Diesmal zählst du von oben nach unten die Anzahl dunkler Karten und ergänzt wieder zu einer geraden Anzahl.



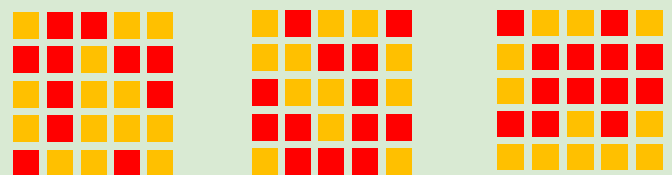
Bitte nun jemanden aus dem Publikum **genau eine** der 25 Karten umzudrehen (hell/dunkel), ohne dass du es sehen kannst. Danach schaust du kurz auf das Spielfeld, um sogleich auf magische Weise auf die Karte zu zeigen, die gedreht wurde. Woher wusstest du, welche es war? Dazu zählst du erneut in jeder Zeile und jeder Spalte die dunklen Karten zusammen. Es wird jetzt genau eine Zeile und eine Spalte mit einer *ungeraden* Anzahl dunkler Karten geben. Dort wo sich diese Zeile und diese Spalte kreuzen, wurde die Karte gedreht. (Ist das nicht der Fall, hat die Person aus dem Publikum beim Umdrehen gemogelt. Sie hat entweder mehrere Karten oder gar keine Karte gedreht.)

Übrigens: Du kannst auch mehr Karten verwenden und zum Beispiel mit 5x5 Karten beginnen – das Prinzip bleibt immer gleich. Probiere den Trick mit deinen Eltern oder Geschwistern aus und lasse sie raten, wie er funktioniert ☺

2. Die Idee der Redundanz und Prüfsumme

In der Informatik wird dieses Prinzip als **Redundanz** bezeichnet. Die zusätzliche Zeile und Spalte benötigen wir nur, um Fehler zu finden, sie sind aber nicht Teil der eigentlichen Information. Je nach Verfahren lassen sich Fehler sogar automatisch korrigieren, wie im Beispiel des Zaubertricks. Wusstest du, dass etwas 33% aller Daten auf einer CD nur zur Fehlererkennung und Fehlerkorrektur genutzt werden? Damit lässt sich eine CD trotz leichter Kratzer immer noch ohne Störgeräusche abspielen. Es gibt auch sonst überall redundante Daten im Alltag. Ein weiteres Beispiel sind **Prüfsummen**. Sie werden im Supermarkt beim Einscannen von Barcodes, bei ISBN-Nummern auf Büchern oder bei der Seriennummer von Geldscheinen verwendet. Auch Bankkontonummern enthalten Prüfziffern, damit bei einem Tippfehler nicht so leicht falsche Überweisungen passieren.

Wo wurde hier jeweils eine Karte gedreht?



Worum geht es?

Auf jeder Milchpackung oder Cola-Dose findest du heute Strichcodes (auch Barcode genannt). Die Supermarktkasse scannt diesen Code ein und weiss sofort, um welches Produkt es sich handelt und wieviel es kostet. Wie funktioniert das eigentlich?



Wie funktioniert es?

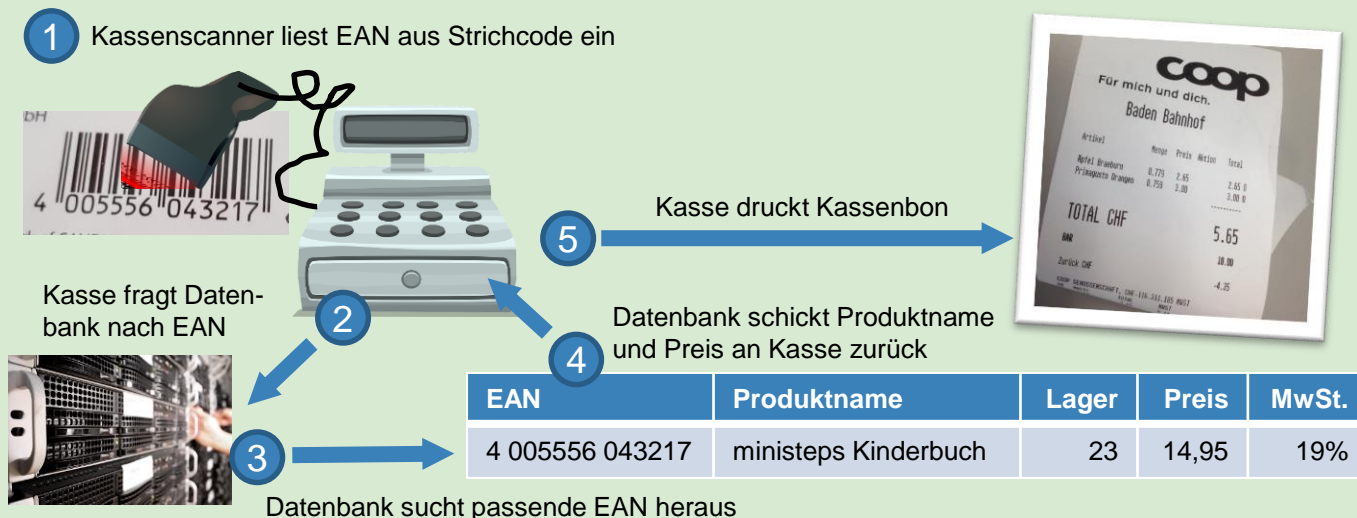
1. Der Strichcode im Supermarkt

Die von der Supermarktkasse bekannten Laserscanner senden Licht auf die schwarzen und weissen Linien des Strichcodes. Schwarz und weiss reflektieren Licht unterschiedlich gut. Das Lesegerät empfängt das reflektierte Licht und kann die Unterschiede zwischen Licht an und Licht aus (oder auch 1 und 0) erkennen. Das passiert mehrere hundert Mal in einer einzigen Sekunde. Die Linien **codieren** dabei nur die Zahl, die unter dem Barcode steht. Die Zahl wird in Europa European Article Number (EAN) genannt. Wenn ein Barcode einmal nicht funktioniert – das hast du sicher auch schon erlebt – dann muss die Zahlenfolge von Hand eingetippt werden.

Die letzte Ziffer ist immer eine **Prüfziffer**, die mit einer komplizierten Rechnung aus den vorherigen Ziffern bestimmt werden kann. Wenn diese Prüfziffer nicht passt, z.B. weil der Code verschmutzt ist, wird die Kasse nicht piepsen.

Im Barcode selbst steckt also weder der Name noch der Preis des Produkts. Hast du eine Idee, warum es unpraktisch wäre, wenn die Preise direkt im Barcode stehen würden?

Im folgenden Diagramm ist dargestellt, wie die Kasse tatsächlich Namen und Preis eines Produkts herausfindet.



Hast du eine Idee, warum in der Datenbank der „Lagerbestand“ aufgeführt wird und was damit passiert, wenn ein Artikel verkauft wird? Sind eigentlich die Barcodes für die gleichen Produkte in allen Supermärkten die gleichen?

Was würde passieren, wenn mehrere Produkte den gleichen Barcode hätten?

2. Aufbau von EAN-Barcodes

Such mal in deiner Umgebung nach Produkt-Barcodes und notiere in einer Tabelle Codes und Produkt.

Fallen Dir Muster auf? Haben diese Muster etwas mit den Produkten zu tun, die du gefunden hast?

40 063811 47095	Leuchtstift gelb
40 063811 47149	Leuchtstift pink
95 560890 07788	Radierer
...	...

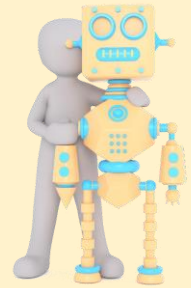
Bei EAN-Nummern stehen die ersten Ziffern für das Land, in dem die Herstellerfirma ihren Hauptsitz hat: 400 bis 440 steht für Deutschland, 760 bis 769 für die Schweiz und 900 bis 919 für Österreich. Es gibt jedoch eine Ausnahme von dieser Regel: Dinge, die gewogen werden müssen (z.B. Früchte und Gemüse). Diese Barcodes fangen immer mit einer 2 an. Im Gegensatz zu normalen EAN beschreibt nur die erste Hälfte das Produkt. Der zweite Teil wird für das gewogene Gewicht verwendet. Über die Produktnummer kann die Kasse den aktuellen Preis pro Kilogramm aus der Datenbank erfragen und diesen mit dem Gewicht aus dem Code multiplizieren. Du kannst eine öffentliche Datenbank wie www.ean-search.org verwenden und deine EAN dort eingeben.

Auf Lösung:

Worum geht es?

Das Programmieren ist ein wichtiger Aspekt der Informatik, wenn auch nicht der einzige. Mit wenigen Grundbefehlen lassen sich komplexe Programme konstruieren, um ein Problem zu lösen.

Das geht auch ohne Computer! Programmier deine Eltern oder Freunde!



Wie funktioniert es?

Auf iLearnIT.ch/stromlos gibt es eine zusätzliche Druckvorlage zu dieser Aufgabe.

1. Roboter initialisieren

Der Computer versteht einfache Befehle, die er stur abarbeitet. Um etwas zu programmieren, benötigt man deshalb ganz präzise Befehle, die nicht missverstanden werden können. Suche dir einen Roboter (Freund/in, Eltern, Lehrer/in ...) der/die dein Programm ausführen und – ganz wichtig – auf keinen Fall deine Fehler oder Ungenauigkeiten selbständig korrigiert. Die folgenden Tafeln zeigen einige mögliche Befehle, die du deinem Roboter erteilen kannst. Du kannst auch eigene Befehle erfinden. Erstelle auf Papier Befehls-Karten mit jeweils einem Befehl als Symbol darauf:



Schritt nach vorn



Schritt zurück

halbe Drehung
nach linkshalbe Drehung
nach rechts

piepsen

runter in
die Hocke

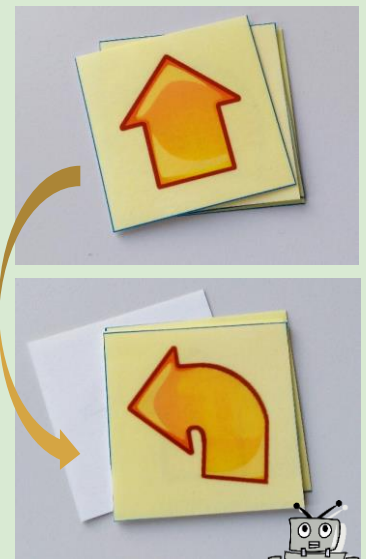
hinstellen

hochheben
(mitnehmen)

ablegen

2. Spielfeld initialisieren und Programmieren

Platziere irgendwo im Raum einen Gegenstand, den der „Roboter“ für dich holen soll. Der „Programmierer“ legt die Befehlskarten nebeneinander, um ein Programm zu schreiben. Ist dein Programm fertig, legt er alle Karten der Reihenfolge nach auf einen Stapel und übergibt den Stapel an den Roboter. Der Roboter beginnt das Programm mit der ersten Karte und führt diese aus. Danach dreht er die Karte auf die Rückseite und legt sie unter den Stapel (siehe Bild). Nun geht es mit jeder Karte so weiter, bis die erste umgekehrte Karte erscheint. Der Roboter führt die Befehle ganz genau aus wie angegeben – natürlich ohne sich selbst oder andere dabei zu verletzen ;-)
Hinweis: Der „Roboter“ kann niemals zwei Dinge gleichzeitig tun. Er wird zum Beispiel erst fertig „piepsen“, bevor er mit dem nächsten Befehl zum Laufen fortsetzt.



3. Wiederholung

Mit einem „Mehrfach-Befehl“ wie 4-Mal oder 10-Mal, lässt sich die unmittelbar danach folgende Befehlskarte wiederholen. Damit kannst du Karten sparen. Wenn eine Karte wiederholt wird, zählt der Roboter laut mit, bevor er die Karte unter den Stapel legt.

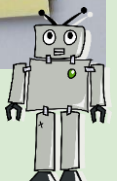
4. Debugging – die Fehlersuche

Läuft dein Roboter zu weit oder zu kurz? Hast du eine Drehung vergessen oder hast du in die falsche Richtung gesteuert? Kein Problem! Stoppe deinen Roboter und lass dir den Stapel zurückgegeben. Verändere dein Programm und versuche es erneut.

Das Suchen und beseitigen von Fehlern nennt man beim Programmieren **Debugging**. Die Bezeichnung **Bug** (dt. Ungeziefer) für einen Fehler in einer Maschine wurde bereits verwendet, als es noch gar keine Computer gab. In den ersten, zimmergrossen Computern war aber durchaus auch mal echtes Ungeziefer der Grund für einen Computerabsturz.

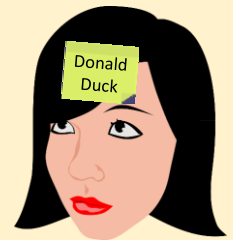
5. Programmtausch und neue Befehle

Sind mehrere Programmierer und Roboter unterwegs, könnt ihr eure Programme (Stapel) auch austauschen und schauen, ob ein anderer Roboter damit ebenfalls zum Ziel gelangt. Da jeder Roboter etwas anders „gebaut“ ist, kann sich zum Beispiel die Schrittlänge unterscheiden. Ihr könnt deshalb auch neue Befehle zeichnen, die mit einer *Abbruchbedingung* verbunden sind. Zum Beispiel: „Laufe nach vorn, bis ein Hindernis kommt“ oder „drehe dich langsam nach links, bis jemand in die Hände klatscht“.



Worum geht es?

Computer können sehr schnell Informationen zum Beispiel in einer Datenbank mit Millionen von Einträgen herausuchen. Damit sie das so schnell können, verwenden sie schlaue Suchstrategien. Diese lassen sich auch ausserhalb der Computerwelt anwenden.

**Wie funktioniert es?****1. Spielt eine Runde: Wer bin ich?**

Für dieses Spiel benötigst du zwei oder mehr Spieler. Alle überlegen sich eine für alle bekannte Figur, Person, Tier oder Gegenstand (dein Lieblingsfussballer ist für die anderen vielleicht total unbekannt, aber z. B. Donald Duck kennen vermutlich alle). Schreibt den Namen eurer Figur verdeckt auf eine Haftnotiz-Zettel. Klebt euren Zettel an die Stirn eines anderen Mitspielers, ohne dass dieser den Namen selbst sehen kann. Ziel des Spiels ist es, den Namen auf seinem Zettel durch geschickte Fragen herauszufinden. Ein Spieler beginnt und stellt Fragen in die Runde, die nur mit Ja oder Nein beantwortet werden können. Bin ich ein Mensch? Bin ich männlich? Bin ich ein Schauspieler? Die anderen Spieler antworten nur mit JA oder NEIN. Sobald eine Frage von der Gruppe mit NEIN beantwortet wird, ist der nächste Spieler an der Reihe. Glaubt der Spieler, der gerade an der Reihe ist, seine Figur herausgefunden zu haben, fragt er „Bin ich Donald Duck?“ und darf sich im Erfolgsfall seinen Notizzettel von der Stirn nehmen. Bei diesem Spiel kommt es auf eine gute Fragestrategie an. Durch welche Fragen kannst du möglichst viele Dinge ausschliessen? Zum Beispiel ist es sinnvoll zunächst allgemeinere Fragen wie „Bin ich weiblich?“ zu stellen, bevor man konkreter wird: „Bin ich eine Prinzessin?“

2. An wen denkst du? – Spiel gegen den Computer

Solltest du gerade ein Smartphone oder einen Computer zur Hand habe, spiele einige Runden mit dem Ratespiel Akinator (<http://de.akinator.com>) und überlege dir, wie das Spiel funktioniert. Wie kann der Computer wissen, an wen du denkst? Wie wählt der Computer wohl seine nächste Frage aus? Schaffst du es, an eine Figur zu denken, die Akinator noch nicht kennt?

**3. Binärsuche – schlau und schnell**

Computer sollen häufig aus Millionen von Informationen eine ganz bestimmte herausuchen.

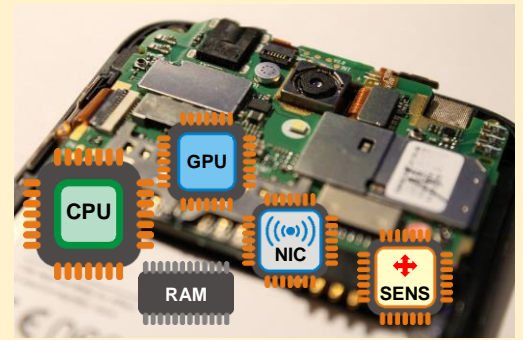
In einer Kundendatenbank soll zum Beispiel Herr „Schmied“ herausgesucht werden. Statt nun jeden einzelnen Eintrag der Datenbank anzusehen und zu prüfen, ob es sich um Herrn Schmied handelt, verwendet die Datenbank einen cleveren Algorithmus. Damit dieser angewendet werden kann, müssen zunächst alle Einträge alphabetisch geordnet werden – wie in einem Telefonbuch. Das Sortieren kann etwas dauern, aber auch dafür gibt es schlaue Algorithmen, die sehr effizient grosse Datenmengen sortieren können. Einmal sortiert, ist die Suche nach Herrn „Schmied“ aber nun viel einfacher. Man teilt den Bestand in zwei gleich grosse Hälften und schaut sich den mittleren Eintrag an. Dort steht zum Beispiel Frau „Pauli“. Jetzt kann man schnell entscheiden, ob Herr „Schmied“ vor oder nach „Pauli“ kommt im Alphabet. In diesem Fall muss die ganze erste Hälfte nicht mehr angeschaut werden, da Schmied dort garantiert nicht dabei sein wird. Mit der verbleibenden Hälfte wiederholen wir das Verfahren. Wieder wird die Auswahl in zwei gleich grosse Teile geteilt und der mittlere Eintrag angeschaut: Herr „Tegel“. Jetzt wissen wir, dass der Eintrag in der ersten Hälfte liegen muss und müssen die zweite Hälfte nicht mehr anschauen. Das setzt sich solange fort, bis Herr Schmied gefunden wurde. Bei einer Millionen Kunden müssen nur höchstens 19 Einträge angeschaut werden, bis garantiert der gesuchte Datensatz gefunden wird. Auch bei 100 Millionen Kunden steigt die Anzahl nötiger Einträge nur wenig an (25). Da man sich bei jeder Entscheidung für eine von zwei Hälften entscheidet, bezeichnet man dieses Verfahren als **Binärsuche**.

4. Die Geschenke des Weihnachtsmanns

Die Wichtel des Weihnachtsmanns haben dieses Jahr wieder Millionen von Geschenken für Kinder auf der ganzen Welt verpackt. Darunter auch 10'000 rote Spielzeugautos mit Fernsteuerung. Nachdem alle Autos identisch verpackt wurden, fällt einem der Wichtel auf, dass eine Fernbediendung übrig geblieben ist. Oh nein, alle Geschenke noch einmal auf packen? Nein nein, sagt ein anderer Wichtel und zeigt auf eine grosse Waage. Wir wiegen die Geschenke einfach ein paar Mal ab und finden so ganz schnell das Auto ohne Fernbediendung. Habt ihr eine Idee, wie die Wichtel das mit nur 15 Mal wiegen schaffen können und nach jedem Wiegen nur noch halb so viele Geschenke überprüfen müssen?

Worum geht es?

Computer werden immer kleiner und passen heute in Form eines Smartphones schon problemlos in die Hosentasche. Ein Computer ist wie ein Lebewesen aus vielen Bestandteilen aufgebaut. Einige Bauteile sind dabei unbedingt nötig, andere können, müssen aber nicht vorhanden sein. In einem Theaterstück spielt ihr selbst ausgewählte Bestandteile eines Smartphones und bekommt einen Eindruck, wie diese zusammenarbeiten.



Wie funktioniert es?

1. Bestandteile kennenlernen

Ein Smartphone oder Tablet ist ein vollständiger Computer in sehr kompakter Bauform. Alle Computer haben die gleichen Grundbestandteile – egal ob Notebook, Spielkonsole oder Server. Schaut euch die nachfolgenden Rollenkarten gemeinsam an und diskutiert die Bedeutung der Bauteile – welche braucht es zwingend, welche nicht?



Der Prozessor

Der Prozessor – auch Central Processing Unit (CPU) genannt – verarbeitet die Befehle der Programme. Häufig wird er auch als das Gehirn des Computers bezeichnet. Seine Stärke liegt im extrem schnellen Berechnen von mathematischen Operationen. Er ist also ein superguter Taschenrechner. Er hat ganz viele kleine Anschlüsse, über die er mit den anderen Bauteilen des Computers verbunden ist. Er steuert von wo Daten gelesen oder wohin sie geschrieben werden. Häufig wird er beim Rechnen richtig heiss. Bei grossen Computern ist deshalb immer ein eigener Lüfter zur Kühlung auf der CPU angebracht.



Der Speicher

Programme, Texte, Fotos oder Videos – alle Daten werden im Speicher abgelegt. Sie können vom Prozessor angefordert werden, wenn sie benötigt werden. Natürlich kann man sie auch löschen. Der besonders schnelle Speicher ist der Arbeitsspeicher (RAM), der als Kurzzeitgedächtnis des Prozessors dient. Mehr Platz hat die langsamere Festplatte im Computer. Smartphones und Tablets haben statt einer Festplatte einen Speicherchip der mehrere Gigabyte an Daten speichern kann. Online-Speicher ist nicht im Gerät selbst verbaut, sondern wird an einem anderen Ort bereit gestellt und übers Netzwerk gelesen und beschrieben.



Die Netzwerkkarte

Dieser Chip ist für das Versenden und Empfangen von digitalen Daten an und von anderen Geräten verantwortlich. Übertragen werden Daten mit einem Kabel, per Funk oder Licht. Verschiedene Übertragungsarten sind dabei unterschiedlich schnell, haben eine grössere Reichweite oder sind energiesparender (z.B. 4G-LTE, WLAN, Bluetooth).



Der Sensor

In heutigen Computern befinden sich viele unterschiedliche Sensoren. Sie messen zum Beispiel ständig die Temperatur. Ein GPS-Chip ermittelt, wo wir uns gerade befinden. Ein Kompass-Sensor und ein Gyrosensor messen, in welche Richtung das Gerät zeigt und so weiter. Alle Messwerte stehen als digitale Daten den anderen Bauteilen zur Verfügung.



Das Display (und Grafikkarte)

Ein Display besteht aus Millionen kleiner Pixel. Jedes Pixel kann in einer anderen Farbe leuchten, wodurch sich ein Bild ergibt. Welches Pixel wann an- oder ausgeschaltet werden soll, bestimmt die Grafikkarte (GPU). Touch-Screens können auch auf Berührung reagieren und die Touch-Position als digitale Daten anderen Bauteilen mitteilen.



Die Kamera

Über eine Linse wird das Licht auf einen Sensor-Chip (CMOS) gelenkt. Er reagiert wie unsere Augen auf einfallendes Licht. Damit kann er Farben und Helligkeit erkennen und als digitale Daten bereitstellen. Andere Bauteile können diese Daten verwenden, um zum Beispiel ein Foto oder Video zu speichern oder eine Person im Kamerabild zu erkennen.

2. Bühnenstück vorbereiten

Im Stück geht es um einen Dialog zwischen den Bauteilen eines Smartphones.

Kursiv geschriebene Texte stehen für Handlungen. Ihr benötigt neben den 6 Schauspielern/innen einige Requisiten.

- Nachricht-Zettel und Antwort-Zettel, Nachricht-Kopie-Zettel und Antwort-Kopie-Zettel (andere Papierfarbe) und einen Stift für den Speicher zum Aufschreiben.
- Kostüme, bei dem das eigene Bauteil für das Publikum ersichtlich wird.
- Eventuell: Smartphone als grosser Deko-Rahmen als Bühnenbild, in dem die Schauspieler stehen können.

Auf iLearnIT.ch/stromlos gibt es eine zusätzliche Druckvorlage zu dieser Aufgabe.



3. Das Stück: Nachricht von Mutti

ALLE: *alle schlafen, schnarchen und stehen sehr eng zusammen*



Allgemeine Kontrolle, alle einmal kurz aufwachen!
Sind alle noch da und gibt es etwas zu tun?
Kamera - was ist draussen los?



Alles ruhig und dunkel, ich glaube wir liegen
verkehrtherum auf dem Tisch, könnte aber auch
Nacht sein.



Hmm unsere Uhr sagt mir es ist 9:30.
Lagesensor - was ist deine Meinung?



Keine Beschleunigung festgestellt.
Ausrichtung sagt waagrecht. Wir liegen herum.



Ok, dann liegen wir wohl auf dem Tisch.
Netzwerkkarte, wie sieht es bei dir aus?
Irgendwelche News aus dem Internet?



Nö nix. Moment, ich glaub da kommt grad was rein
ratter ratter Wir haben eine Nachricht von der
Nummer 0815110 bekommen. Was soll ich machen?



Erstmal Speichern, damit nichts verloren
geht, wenn uns der Akku ausgeht. Speicher?
Schreib schon mal auf von wem sie kommt.



Ok, ich notiere Nachricht von 0815110 kommt da
noch mehr?



Ja, Moment. Netzwerkkarte, gib mir bitte den
Inhalt der Nachricht.



reicht Nachricht-Zettel an den Prozessor
Hier bitte schön.



Ok, Speicher - schreib das mal auf.
gibt Zettel an Speicher weiter



nimmt Zettel entgegen Ok ich schreibe auf:
Ein M ein I ein T, noch ein T, A ein G, E, S, S, E N.
Murmelt noch etwas leise weiter vor sich hin



Display? Mach schon mal das Licht an.
Wir müssen den User benachrichtigen.



*macht ein *bling* Licht ist an, was soll ich darstellen?*



Hey Speicher, wie war nochmal die Nachricht?
Ich muss sie ans Display weiterleiten.



gibt den Nachricht-Kopie-Zettel an den Prozessor
Hier eine Kopie für dich, das Original bleibt bei mir.



Dankeschön.
gibt Zettel an Display weiter



MiiiiTTAAAGGGGEESSSEEEENN DAHEIM
Fragezeichen, MUTTI. Ok, hab ich aufs Display in
Schönschrift in einen Benachrichtigungskasten
geschrieben.



Super danke. Wenn wir auf dem Tisch liegen
bekommt das der User aber eh nicht mit.
Wir brauchen Vibration.

ALLE: *surren und wackeln 3x hintereinander*



Hey Kamera, kannst du das bestätigen?



Hey Leute, da passiert was - ich messe grössere Bewegung, ich glaube wir werden grad angehoben.



Ok, Kaffeepause, das dauert jetzt, bis der User die Nachricht gelesen und verstanden hat – diese Menschen sind so unglaublich langsam.



Ja, es wird wieder hell und ich sehe ein Gesicht.



Ok, alle wieder aufwachen, es gibt Arbeit.



Leute - ich spüre eine Berührung. *windet sich etwas* Ich werde betoucht. *kichert* Das kitzelt.



Moment, ich starte das Nachrichten-Programm mit dem Antwortfeld. Zeig schon mal die Tastatur an, damit der User schreiben kann.



Der User hat auf "Antworten" gedrückt. Was jetzt?



Gut gut. Melde mir, sobald der User Buchstaben drückt.



Ok, Roger - ich zeichne die Tastatur auf die untere Bildschirmhälfte.



Moment. Speicher? Schreib mal mit, was das Display da bekommt.



duckt sich bei jedem Buchstaben zusammen hier kommt ein J und ein A.



Ok, ich notiere.
Nimmt Antwort-Zettel in die Hand und schreibt mit.



Hey Netzwerkkarte, kannst du bitte eine Nachricht per Funk an 0815110 zurückschicken?



duckt sich bei jedem Buchstaben etwas zusammen
G E R N E Komma H A B E H U N G E R
duckt sich zweimal stark „SENDE, SENDE!“



Speicher, kannst du mir bitte die Antwort vom User geben?



Ja, kann losgehen - was soll ich senden?



gibt den Zettel an Netzwerkkarte weiter Hier, bitte senden.



gibt einen Antwort-Kopie-Zettel an Prozessor Klar doch, hier hast du eine Kopie davon.



Keine Angst, Ist ja nur eine kurze Nachricht – sind wir lieber froh, dass der User grad kein Video aus dem Internet anschauen will.



Sende Daten ... *stöhnt laut*
Das ist soooo anstrengend. Wenn ich das noch lange mache, geht uns gleich wieder der Akku aus.



Hey, ich glaub wir werden gerade wieder abgelegt. Ich spüre richtig viel Bewegung.



Alles gut, wir können uns wieder schlafen legen. Ich wecke euch in ein paar Minuten wieder für einen Statusbericht.



Oh nein, jetzt wird es auch schon wieder dunkel und ich sehe nichts mehr.

ALLE: *alle schlafen ein, schnarchen und stehen wieder sehr eng zusammen*



Hach, das ist sooo eng hier drin. Wisst ihr Leute, früher gab es noch so richtig grosse Computer unter dem Schreibtisch. Da war noch Platz sag ich euch. Aber heute muss ja alles in so ein kleines Gerät passen, dass man uns nicht mal mehr richtig auseinanderhalten oder reparieren kann.

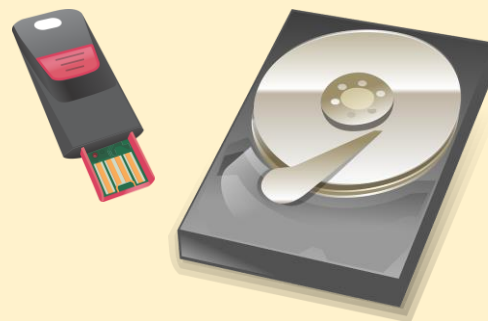
Idee: Zum Vergleich kann man einen alten Computer und seine Bauteile einmal genauer anschauen.



Wieviel ist eigentlich ein Gigabyte?

Worum geht es?

Texte, Fotos, Videos, Spiele und Programme werden auf der Festplatte oder auf einem Speicherchip abgespeichert. Doch auch wenn die Computer jedes Jahr mehr Speicherplatz haben, kann der Speicher auch einmal voll sein. Dann muss man aufräumen und nicht mehr benötigte Dinge löschen. Aber was löscht man dann am besten, damit wieder ausreichend Speicherplatz zur Verfügung steht? Was gibt es eigentlich für Daten auf deinem eigenen Gerät?



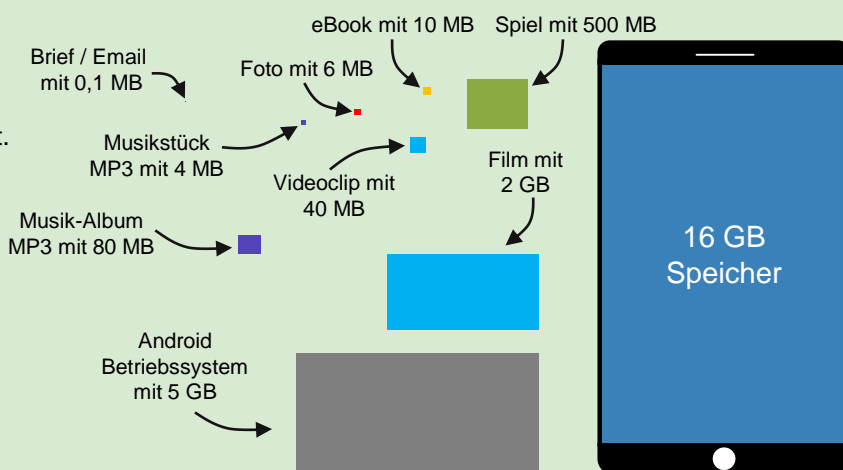
Wie funktioniert es?

Auf iLearnIT.ch/stromlos gibt es eine zusätzliche Druckvorlage zu dieser Aufgabe.

1. Speicherplatzbedarf von Daten

Finde heraus, wieviel Speicherplatz dein Computer, Tablet oder Smartphone hat. Vermutlich wirst du eine Zahl mit der Einheit GB für Gigabyte finden. Doch was muss man sich unter einem Gigabyte vorstellen? Das Smartphone auf diesem Blatt hat 16 GB Speicherplatz. Das Display entspricht einer Fläche von 16 cm². Daneben sind einige Dateien mit ihrem typischen Speicherplatzbedarf ebenfalls als Flächen dargestellt. Kannst du auf deinem Gerät herausfinden, welche Dateien den meisten Speicherplatz belegen? Das Betriebssystem verbraucht zwar viel Speicherplatz bei einem Smartphone, ist aber zwingend nötig und kann deshalb nicht gelöscht werden. Du solltest immer nur Dateien löschen, von denen du weisst, wozu sie gut sind.

Videos, Spiele und Fotosammlungen sind meist die grössten Speicherplatzfresser. Es lohnt sich also kaum, wenn du Briefe oder deinen letzten Aufsatz löschst, wenn du Speicherplatz brauchst.



Umrechnungstabelle

- 1 Bit = genau eine 0 oder eine 1
- 1 Byte = 8 Bit
- 1 Kilobyte = 1024 Byte
- 1 Megabyte = 1024 Kilobyte
- 1 Gigabyte = 1024 Megabyte
- 1 Terabyte = 1024 Gigabyte
- 1 Petabyte = 1024 Terabyte

2. Datentypen und Dateierweiterungen kennenlernen

Für jede Art von Daten (Videos, Musik, Texte usw.) gibt es verschiedene, gebräuchliche *Datenformate*. Diese beschreiben genau, wie die Daten nacheinander in 0en und 1en aufgeschrieben werden müssen, damit verschiedene Programme sie verarbeiten können. Auf Blatt 5 hast du bereits ein eigenes Datenformat für Klebezettel erfunden. Für Dokumente gibt es zum Beispiel das Open-Document-Format mit der Dateierweiterung **.odt**. Ein Scratch-Projekt wird im Scratch-File-Format mit der Endung **.sb** oder **.sb2** gespeichert. Welche anderen Dateiformate sind dir schon begegnet? Schaut auf euren Geräten Dateien und Programme an und erstellt eine Tabelle. Dateierweiterungen werden nicht immer in der Grundeinstellungen des Geräts angezeigt. Finde mit Hilfe einer Suchmaschine eine Möglichkeit diese einzublenden.

3. Dateierweiterungen - Experiment

Das Betriebssystem deines Computers kann an der Dateierweiterung erkennen, mit welchem Programm eine Datei geöffnet werden soll. Klickst du eine **.docx** Datei an, öffnet sich dein Textverarbeitungsprogramm, eine **.pptx** Datei öffnet deine Präsentationssoftware usw. Lade ein Bild von pixabay.com herunter und speichere es auf deinem Gerät ab. Finde eine Möglichkeit, die Dateierweiterung in **.txt** umzubenennen. Öffnest du nun die Datei, sollte ein einfacher Texteditor gestartet werden. Kannst du dir das Ergebnis erklären? Vergleiche eure Ergebnisse untereinander. **Achtung:** Das Umbenennen der Dateierweiterung verändert nicht das Datenformat. Dafür braucht es spezielle Konvertierungsprogramme.



cat.jpg.txt

Datenart	Programm	Endung
Dokument	Word	.docx
Präsentation	PowerPoint	.pptx
Programm	Scratch 2.0	.sb2
Musik		
Foto		
Video		

Worum geht es?

Viele Konzepte der Informatik lassen sich auch ohne Hilfe von Computern, dafür mit Hilfsmitteln wie Papier, Stiften, Würfeln oder Spielkarten erklären bzw. im wahrsten Sinne des Wortes „be-greifen“.

Solche Erklärungen werden als „Computer Science Unplugged“ (auf deutsch in etwa „Informatik ohne Strom“) bezeichnet und finden immer grössere Verbreitung. Mit der vorliegenden Broschüre sind einige, teilweise aufeinander aufbauende Beispiele nun auch auf deutsch verfügbar.

Was wird benötigt?

- Spielkarten
- Papier
- Stifte
- Haftnotizen
- In der Regel kein Computer!

Was lässt sich damit zeigen?

Neben den konkreten Konzepten lassen sich mit Computer Science Unplugged auch folgende Aussagen implizit vermitteln:

- Informatik ≠ Computer
- Informatik ≠ Programmieren
- Informatik ≠ schnelllebig
- Informatik lässt sich auch einfach erklären

Warum Informatik ohne Strom?

Computer Science unplugged bietet verschiedene Vorteile:

- Konzepte können im wahrsten Sinne des Wortes „begriffen“ werden
- Bestehende (positive und negative) Vorurteile gegenüber Computern fallen weg
- Es wird keine teure Infrastruktur benötigt
- Die Beispiele funktionieren auch nach einem Jahr noch ohne Update
- Der Vorbereitungsaufwand ist meist geringer
- Es geht keine Zeit mit Produktdetails verloren
- Die Beispiele funktionieren unabhängig davon, ob gerade wieder eine neue Softwareversion installiert wurde oder das WLAN streikt.

Zitat

„In der Informatik geht es genau so wenig um Computer, wie in der Astronomie um Teleskope.“

Edsger W. Dijkstra

Links & Literatur

www.csunplugged.org



Impressum / Kontakt

Download: <http://iLearnIT.ch/stromlos>
Pädagogische Hochschule Schwyz
Michael Hielscher / Beat Döbeli
michael.hielscher@phsz.ch
beat.doebeli@phsz.ch

Weitere Broschüren zum Thema Informatik

Die Pädagogische Hochschule Schwyz hat mehrere Broschüren zum Thema Informatik erarbeitet, die frei verfügbar sind:

Makey Makey Board

(Nicht nur) mit Bananen Musik machen:



iLearnIT.ch/makey

Ozobot

Programmieren mit Filzstiften:



iLearnIT.ch/ozobot

Scratch Gruppenpuzzle

Gemeinsam Spiele programmieren:



iLearnIT.ch/puzzle