

Wie verarbeitet ein Computer Zeichen?

Ein Computer kennt an sich nur Bits, also die Ziffern 1 und 0. Indem man mehrere Ziffern (Bits) zusammenfasst, kann man größere Zahlen darstellen (genau wie in unserem Zahlensystem, dem Zehnersystem, auch). Eine Zahl aus 8 Bits heißt beispielsweise Byte. Bei festgelegter Bitanzahl schreibt man auch führende Nullen mit auf.

Buchstaben kennt der Computer ebenfalls von Natur aus nicht. Sie müssen daher als Zahlen gespeichert werden. Welche dieser Zahlen welches Zeichen darstellen, dafür gibt es verschiedene Zuordnungen, genannt Zeichensätze. Möglicherweise hast du schon mal erfahren müssen, was passiert, wenn Zeichensätzen von Sender und Empfänger nicht übereinstimmen. Nur die Zahlen 0 bis 127 entsprechen fast in allen Zeichensätzen den gleichen Buchstaben, die man 1968 als ASCII (*American Standard Code for Information Interchange*) festgelegt hat und bis heute so nutzt. 0 bis 31 und 127 sind allerdings keine Buchstaben; 32 ist das Leerzeichen.

Auf dem Gerät, das *keine* grüne LED hat, kannst du eine Zahl einstellen, indem du die Hebelchen des blauen Kastens bewegst. Ist der linke Hebel unten, wird die Zahl um 128 erhöht (wird hier nicht verwendet, da alle Zahlen unter 128 sein sollen), beim zweiten von links um 64, dann 32, 16, 8, 4, 2 und ganz rechts um 1. Jeder Hebel entspricht einer der acht LEDs darüber. Eine leuchtende LED steht für eine 1, ausgeschaltete für eine 0. Leuchten also z.B. nur die rechten 6 LEDs, entspricht das der Binärzahl 00111111, oder 63 im Dezimalsystem (siehe dazu die Tabelle auf der nächsten Seite). Letzteres wird unten im Display angezeigt, in der Zeile darüber der dazugehörige Buchstabe, in diesem Fall das Fragezeichen. Der Taster neben der LED schickt das Zeichen zum Gerät mit der grünen LED.

Aufgabe 1: Jeder überträgt seinen Namen mit den Geräten. Notiere dir die Zeichen deines Namens im Binärsystem (basierend auf den acht LEDs über den Hebelchen) und Dezimalsystem (wird im Display angezeigt) und ergänze die Tabelle auf der nächsten Seite um die fehlenden Zeichen (Nummer 64 bis 91 und 97 bis 123). Da gibt es sicher auch ein System.

Beispiel: Ich (der dieses Arbeitsblatt gemacht hat) heiße Janni. Das wäre dann 01001010 01100001 01101110 01101110 01101001 im Binär- und 74, 97, 110, 110, 105 im Dezimalsystem.

Dein Name im Binär- und Dezimalsystem:

Aufgabe 2: Welche Zusammenhänge fallen dir auf und was könnte der Grund dafür sein? Wie unterscheiden sich beispielsweise Groß- und Kleinbuchstaben in ihrer Kodierung?

Aufgabe 3: Warum fehlen die Sonderzeichen wie Ä, Ö, Ü und ß im ASCII-Zeichensatz?

Aufgabe 4: Welche Probleme können sich daraus ergeben, dass diese Zeichen fehlen?

	Dez	Binär	Dez	Binär	Dez	Binär
<i>leer</i>	32	00100000	64	01000000	`	96 01100000
!	33	00100001	65	01000001		97 01100001
"	34	00100010	66	01000010		98 01100010
#	35	00100011	67	01000011		99 01100011
\$	36	00100100	68	01000100		100 01100100
%	37	00100101	69	01000101		101 01100101
&	38	00100110	70	01000110		102 01100110
'	39	00100111	71	01000111		103 01100111
(40	00101000	72	01001000		104 01101000
)	41	00101001	73	01001001		105 01101001
*	42	00101010	74	01001010		106 01101010
+	43	00101011	75	01001011		107 01101011
,	44	00101100	76	01001100		108 01101100
-	45	00101101	77	01001101		109 01101101
.	46	00101110	78	01001110		110 01101110
/	47	00101111	79	01001111		111 01101111
0	48	00110000	80	01010000		112 01110000
1	49	00110001	81	01010001		113 01110001
2	50	00110010	82	01010010		114 01110010
3	51	00110011	83	01010011		115 01110011
4	52	00110100	84	01010100		116 01110100
5	53	00110101	85	01010101		117 01110101
6	54	00110110	86	01010110		118 01110110
7	55	00110111	87	01010111		119 01110111
8	56	00111000	88	01011000		120 01111000
9	57	00111001	89	01011001		121 01111001
:	58	00111010	90	01011010		122 01111010
;	59	00111011	91	01011011		123 01111011
<	60	00111100	\	92 01011100		124 01111100
=	61	00111101] 93	01011101	}	125 01111101
>	62	00111110	^ 94	01011110	~	126 01111110
?	63	00111111	_ 95	01011111		