

Aufgabe 1

Man muss zum ASCII-Code eines Grossbuchstabens die die Zahl 32 addieren, um den entsprechenden Kleinbuchstaben zu erhalten.

Aufgabe 2

33 Steuerzeichen (Die Zeichen mit den Nummern 0 bis 31 und 127)

Aufgabe 3

$0xA9 = 10101001_2$

Aufgabe 4

$0x1D11E = 0x0001D11E$
 $= 0000\ 0000\ 0000\ 0001\ 1101\ 0001\ 0001\ 1110$

Aufgabe 5

1. Zeichen 11101111 10000000 10111111
2. Zeichen 01011010
3. Zeichen 11011010 10100011

Also besteht der Bytestrom aus drei Zeichen.

Aufgabe 6

11100001 10001110 10101000

Codierungsinformation entfernen:

00001 001110 101000

Bits von rechts in 4er-Gruppen zusammenfassen:

0001 0011 1010 1000

Ins Hexadezimalsystem umrechnen:

0x13A8

Nebenbei: 0x13A8 repräsentiert den Cherokee-Buchstaben GE.

Aufgabe 7

0x1F0BD in Binärform:

1 1111 0000 1011 1101

Es müssen 17 Bits codiert werden. Wir erinnern uns:

| | |
|-------------------------------------|-----------|
| 0xxxxxxx | 7 Bit |
| 110xxxxx 10xxxxxx | 8–11 Bit |
| 1110xxxx 10xxxxxx 10xxxxxx | 12–16 Bit |
| 11110xxx 10xxxxxx 10xxxxxx 10xxxxxx | 17–21 Bit |

Es sind 4 Bytes nötig: (fehlende Stellen links mit Nullen auffüllen)

11110000 10011111 10000010 10111101

Aufgabe 8

Die Hexadezimalzahl 0x20AC kann durch $4 \cdot 4 = 16$ Bits dargestellt werden.

Umrechnung von 0x20AC ins Binärsystem:

0010 0000 1010 1100

Aufgabe 9

Da 0x1D11E grösser als 0xFFFF ist, benötigt man eine 32-Bit-Darstellung.

Dazu subtrahiert man $2^{16} = 0x10000$ von der Unicode-Nummer:

$0x1D11E - 0x10000 = 0xD11E$

Diese Zahl wandelt man ins Binärsystem um und füllt, falls nötig, bis zur Länge von 20 Bit links mit Nullen auf:

0000 1101 0001 0001 1110

Schliesslich setzt man vor jede 10-Bit-Gruppe das jeweilige Präfix und stellt die 4er-Blöcke im Hexadezimalsystem dar:

1101 1000 0011 0100 1101 1101 0001 1110

D8 34 DD 1E