UTF-8

Author: Philipp G. Freimann (BBW (Berufsbildungsschule Winterthur) https://www.bbw.ch)

Um alle Zeichen lebender Sprachen in ein und derselben Kodierung unterzubringen, wurde **Unicode** (www.unicode.org) geschaffen. Die meisten Programme und Betriebssysteme sind heute in der Lage, Unicode-Zeichen zu verarbeiten und korrekt darzustellen. Der ursprüngliche Unicode verwendet 16 Bits pro Zeichen; damit sind 2^{16} = 65 536 Symbole darstellbar. Der aktuelle Unicode (Version 6.0) verwendet sogar 20 Bits, doch sehr viele Plätze sind noch nicht vergeben. Die meisten Texte (Deutsch, Englisch, Französisch) kommen jedoch mit viel weniger Zeichen aus. Daher entstand die Idee, diese häufigsten Zeichen auch nur mit 8 Bits (= ein Byte) darzustellen.

Die UTF-8 Kodierung -- die obige Komprimierung umsetzt -- ist wie folgt definiert:
Zeichen mit einem (Unicode) Wert < 128 werden in einem Byte gespeichert (dabei bleibt das erste Bit immer auf 0 (Null): 0xxx xxxx.
Zeichen mit einem Wert von 128 bis 2047 werden in zwei Byte gespeichert: 110x xxxx, 10xx xxxx. Dabei beginnt das erste Byte mit zwei Einsen gefolgt von einer Null und fünf Bits für die "Nutzdaten". Das zweite Byte startet mit "10" und sechs Nutzdaten-Bits. Somit sind elf Nutzdaten-Bits vorhanden. Damit können 2^{11} Zeichen dargestellt werden.
Analog verhält es sich mit Zeichen im Bereich 2048 bis 65 535. Diese werden in drei Byte mit sechzehn Nutzdaten-Bits wie folgt gespeichert: 1110 xxxx, 10xx xxxx, 10xx xxxx.

Diese Aufgabe ist ohne Programmieren lösbar: Welche Zeichen verbergen sich hinter den folgenden Zahlen? Schreiben Sie die Zahlen zunächst binär (mit je 8 Binärziffern) auf, suchen Sie anschließend die "Nutz-Bits" und errechnen Sie daraus die Zahl des kodierten Zeichens, das Sie abschließend unter www.unicode.org nachschlagen.
64
80
194, 190
195, 139
239, 172, 129
226, 137, 136