Zeichendarstellung mit Unicode und UTF-8

ASCII – American Standard Code for Information Interchange

In der Mittelstufe wurde die Codierung von Text mithilfe des ASCII-Standards besprochen. Hierbei wird jedem Zeichen ein Wert zwischen 0 und 255 (8 Bit) zugewiesen. Oben siehst du die ASCII-Codetabelle, leere Zellen enthalten Steuerzeichen, welche für die Darstellung am PC nötig waren. Die wichtigsten Steuerzeichen sind in der Tabelle beschrieben.

In einem früheren, hauptsächlich in Amerika benutzten Standard waren lediglich die Zeichen von 0 bis 127 definiert, das letzte, achte Bit wurde zur Fehlerüberprüfung verwendet. Erst später wurde das 8. Bit dazu genommen, um weitere Zeichen, wie z.B. die deutschen Umlaute codieren zu können.

(A1)

Wandle die nachfolgenden Wörter, die in Hexadezimal-Darstellung vorliegen, in lesbaren Text um:

1. `49 6E 66 6F 72 6D 61 74 69 6B`
2. `42 69 6E E4 72`
3. `43 6F 6D 70 75 74 65 72`

Mit einer 8-Bit-Codierung lassen sich nicht mehr Zeichen darstellen, was insbesondere bei anderen Sprachen – wie z.B. griechisch – andere Codierungen nötig machte. Da in diesen Sprachen jedoch die bei uns gebräuchlichen Umlaute nicht benötigt werden, wurde der durch das 8. Bit hinzugekommene Block vom Zeichen 128 bis 255 für die dortigen Zeichen verwendet. Diese und andere länderspezifischen Codierungen lassen sich z.B. unter https://de.wikipedia.org/wiki/ISO_8859 nachschauen.

(A2)

Welche der obigen Wörter würden mit den griechischen Zeichensatz falsch dargestellt werden und warum?

Unicode - UTF-8

Um Probleme, die sich zum einen mit unterschiedlichen Zeichensätzen, zum anderen auch durch andere Sprachen, die mehr als 128 Zeichen umfassen, ergeben haben, wurde der Unicode-Standard entwickelt. Hier kann ein einzelnes Zeichen in der UTF-8-Codierung bis zu 4 Bytes umfassen, nach folgenden Regeln:

- Ist die Binärdarstellung des Unicode-Codes nicht länger als ein Byte und das das erste Bit eine 0, werden die restlichen 7 Bit gemäß des ASCII Codes verwendet, die 128 verbleibenden Möglichkeiten entsprechen also genau dem ASCII-Code.
- Ist die Binärdarstellung des Unicode-Codes länger als ein Byte oder der Code ist ein Byte lang und beginnt mit einer 1 geht man wie folgt vor: Der Unicode-Code wird in 6 Bit lange Teile aufgeteilt. Für jedes dieser 6 Bit Pakete wird ein Byte zur Darstellung verwendet, jedes Byte beginnt mit '10'. Das erste Byte beginnt mit einer '1' für jedes Byte, das verwedndet wird. Benötihgt man also 3 Byte, um ein Zeichen in UTF-8 darzustellen, beginnt das erste Byte mit '111'.

Beispiele:

$y = 79_{16} = 0111\ 100_2$

Beginnt mit einer Null und ist nicht länger als ein Byte → die letzen 7Bit werden verwendet, um zu codieren, also ein „ASCII k“ in UTF-8

UTF-8: 0110 1011