

Tips & Tricks für Einsteiger

Diesmal wollen wir den Neulingen unter den C 64-Fans eine Lösung zum größten Problem aller Anfänger anbieten: Ein leicht verständliches Verfahren zur Umrechnung von dezimalen und hexadezimalen Zahlen.

Immer mehr Leser fragen uns, wie sie möglichst einfach zwischen den beiden Zahlensystemen Dezimal und Hexadezimal umrechnen können. Wir wollen versuchen, einen leicht verständlichen Lösungsweg zu erarbeiten.

1. Was ist das Hexadezimal-System?

In unserem normalen Zehnersystem repräsentiert jede Stelle einer Zahl eine Zehnerpotenz. Ein Beispiel: Die Zahl 4714 läßt sich auch als Summe von Zehnerpotenzen schreiben. $4714 = 4 \times 10^3 + 7 \times 10^2 + 1 \times 10^1 + 4 \times 10^0 = 4 \times 1000 + 7 \times 100 + 1 \times 10 + 4 \times 1$

Beim Hexadezimalsystem wird nun jede Stelle einer Zahl nicht mehr durch eine Zehner-, sondern durch eine Sechzehnerpotenz repräsentiert. Auch hier wieder ein Beispiel: Die Hexadezimalzahl 0324 bedeutet nichts anderes als $3 \times 16^2 + 2 \times 16^1 + 4 \times 16^0 (= 3 \times 256 + 2 \times 16 + 4 \times 1)$.

Dies hat aber noch weitere Konsequenzen:

Im Dezimalsystem wird eine Stelle immer von 0 bis 9 (insgesamt 10 Ziffern) durchgezählt, bevor die nächste Stelle um eins erhöht wird. Also

00 01 02 03 04 05 06 07 08 09 10...

Im Hexadezimalsystem jedoch wird eine Stelle um sechzehn Werte erhöht, bevor zur nächsten Stelle ein Wert hinzugefügt wird. Da aber unsere Ziffern von 0 bis 9 dazu nicht ausreichen, wurden zusätzlich die Buchstaben A bis F herangezogen. Sie vertreten die Zahlenwerte 10 bis 15 (von 0 bis 15 sind es 16 Werte!). Es bedeuten:

A = 10, B = 11, C = 12, D = 13, E = 14, F = 15

Die Dezimalzahl 10 ist also gleichwertig mit dem Hexadezimalwert OA. Damit wären wir auch schon bei der Umrechnung.

2. Dezimal — Hexadezimal

Wenn wir eine Dezimalzahl in Hexadezimal umrechnen wollen, so bauen wir den Hexwert Stelle für Stelle von links nach rechts auf.

Nehmen wir also an, wir möchten die Dezimalzahl 41717 in Hexadezimal umrechnen. Dazu teilen wir sie erst einmal durch 16^3

$$41717 : 16^3 = 10,1848145$$

Uns interessiert hier nur die Vorkommastelle 10. Sie ist gleichbedeutend mit dem Hexwert A. Er bildet die letzte Stelle unserer Hexadezimalzahl.

Nun müssen wir von unserer Dezimalzahl 10×16^3 abziehen. Also $41717 - 10 \times 16^3 = 757$

Um die nächsten Stellen unserer Hexzahl zu erhalten, führen wir diese Prozedur nun noch mit 16^2 und 16^1 durch:

$$757 : 16^2 = 2,9703125 (=2)$$

$$757 - 2 \times 16^2 = 245$$

$$245 : 16^1 = 15,3125 (=F)$$

$$245 - 15 \times 16^1 = 5$$

Als endgültige Umrechnung der Zahl 41717 ins Hexadezimalsystem erhalten wir also **A2F5**.

3. Hexadezimal — Dezimal

Diese Umrechnung ist schon wesentlich einfacher. Um die Hexzahl A2F5 wieder zurückzurechnen, geht man wie folgt vor:

$$Ax16^3 + 2x16^2 + Fx16^1 + 5x16^0$$

Da man aber mit den Buchstaben A und F nicht rechnen kann, müssen diese als Dezimalzahlen angegeben werden. $10 \times 16^3 + 2 \times 16^2 + 15 \times 16^1 + 5 \times 16^0$

Wenn Sie dies auf Ihrem C 64 einmal ausrechnen, so werden Sie als Ergebnis wieder die Zahl 41717 erhalten!

Als Abschluß unseres kleinen Kurses könnten Sie einmal versuchen, ein Basic-Programm zu schreiben, das diese Berechnungen ausführt.

(tr)

»NEW« rückgängig machen

Wie schnell hat man einmal »NEW« eingetippt, um erst hinterher festzustellen, daß man das Programm noch gar nicht gespeichert hat. Mit dieser kleinen Routine können Basic-Programme nach einem »NEW« wieder vollständig zurückgeholten werden.

Man geht dabei so vor: Zuerst einmal muß das Listing 1 eingetippt und gestartet werden. Es erzeugt dann auf Diskette das Programm »AUTO-OLD«. Hat man nun einmal aus Versehen »NEW« eingegeben, so legt man die Diskette mit dem Old-Programm in die Floppy und gibt »LOAD "AUTO-OLD", 8,1« ein. Das Old-Programm wird nun geladen und automatisch gestartet. Ein eventuell gelöschtes Basic-Programm ist wieder hergestellt.

(Georg Brandt/tr)

```
0 :REM ' AUTO-OLD BY G. BRANDT ' <099>
10 OPEN 1,8,1,"AUTO-OLD.P.W":FOR I=0 TO 40
  :READ A:PRINT#1.CHR$(A):NEXT:CLOSE 1 <059>
20 DATA 221,2,165,44,160,1,145,43,32,51,16
  5,165,34,24,105,2,133,45,165,35,105 <068>
30 DATA 0,133,46,169,131,162,164,141,2,3,1
  42,3,3,108,2,3,139,227,221,2 <079>
```

Listing 1. Die Auto-OLD-Routine können Sie mit Hilfe des Checksummers eingeben.

Berechnung des Wochentages

Der nachfolgend hergeleitete Einzeler berechnet den genauen Wochentag innerhalb der Jahre 1900 und 2099 nach Eingabe des Datums (Tag, Monat, Jahr) und gibt ihn mit 0=Sonntag, 1=Montag, 2=Dienstag, etc. an.

6 INPUT T,M,J : F% = T-4 + INT (M/.39) + J/.8 + (M>2) * 9/4: PRINT

Herleitung:

Zunächst wird die Anzahl der Tage, die seit dem eingegebenen Datum und dem 01.01.0001 verstrichen sind, ermittelt.

Diese erhält man durch den eingegebenen Tag, plus die Tage der Vormonate innerhalb des eingegebenen Jahres durch die Teilformel $Mx7-31-INT(Mx0,43) + (M>2)x2$. Hinzugaddiert werden die Tage aller Jahre zuvor (berechnet durch $+(J-1)x365$, die Schalttage werden hinzugezählt). Es gilt $+INT((J+(M>2))/4)$. Der Teil $(M>2)$ verhindert, daß in einem Jahr mit Schalttag vor dem 28. Februar ein Tag zuviel hinzugaddiert wird. Nach dem Gregorianischen Kalender (das Jahr mit 365,2425 Tagen) werden alle 400 Jahre 3 von 4 Schalttagen ausgelassen. Dies sind die nicht durch 400 teilbaren Säkularjahre. Damit gilt $-INT(J/100) + INT(J/400)$.

Man erhält die Gesamtzahl der Tage bis zurück zum Beginn der Zeitrechnung. Nun muß berechnet werden, wie oft in die ermittelte Zahl die 7 hineinpaßt. Der übrige ganzzahlige Rest gibt dann den Wochentag an.

(Frank Kronz/tr)