

# Programmieren C: Bitoperationen: Ausgabe von Zahlen binär und hexadezimal

## *Klaus Kusche*

Schreib ein Programm, das alle auf der Befehlszeile angegebenen **int**-Zahlen der Reihe nach als 32-stellige Binärzahl (mit führenden Nullen) ausgibt.

### Hinweise:

- Verwende zur Umwandlung in binär und zur Ausgabe weder **printf**-Formate noch Multiplikation, Division oder Restbildung (einlesen darfst du die **int**'s wie bisher mit **atoi**).
- Extrahiere stattdessen die einzelnen Bits für die Ausgabe der Reihe nach mit Bitoperationen aus dem **int**.
- Ein einzelnes Zeichen c kannst du mit der Funktion **putchar(c)** (aus **stdio.h**) ausgeben, **c** ist der ASCII-Wert des auszugebenden Zeichens, im einfachsten Fall eine **char**-Konstante wie z.B. **'0'**.
- Welchen int-Typ sollte die Variable für die eingelesene Zahl (und auch Hilfsvariablen für die Bitoperationen) haben, damit es bei den Bit-Schiebe-Operationen keine Überraschungen gibt?

### Zusatzaufgaben:

- Kannst du die Binärzahl schön in mit Zwischenräumen getrennten Vierergruppen ausgeben?

Auch das solltest du ohne Multiplikation, Division oder Restbildung schaffen! (welche Bit-Operation kann eine Restrechnung mit 4 ersetzen?)

- Kannst du die eingelesene Zahl auch als 8-stellige Hexzahl ausgeben? (wieder mit führenden Nullen und aufgeteilt in zwei Vierergruppen)

In diesem Fall musst du nicht einzelne Bits aus dem **int** extrahieren, sondern in jedem Schleifendurchlauf vier aufeinanderfolgende Bits gemeinsam.

- Schaffst du es, die auszugebende Ziffer **'0'** oder **'1'** bei der Binär-Ausgabe direkt aus der Zahl zu berechnen, ohne ein **if** oder einen Conditional Operator zu verwenden?

Erinnere dich: Mit **char**-Werten kann man rechnen...

Bei der Hex-Ausgabe solltest du mit einem einzigen **if** oder **?** : für die Unterscheidung zwischen Hex **'0'**-**'9'** und Hex **'A'**-**'F'** auskommen und nicht 16 einzelne Fälle unterscheiden.

Alternativer Trick: Verwende den extrahierten Wert 0 bis 15 als Index in ein Array, das die auszugebenden Zeichen **0123456789ABCDEF** enthält.