

# Praktikum Programmiersprache C - Bitkette

---

Die Programmiersprache C bietet über ihre Bibliotheksfunktionen eine Reihe komfortabler Ausgabekonvertierungen in dezimaler, hexadezimaler oder oktaler Form. Mitunter möchte man aber eine Zahl oder Bitmaske bitweise mit ihren Nullen und Einsen darstellen.

Schreiben Sie ein C-Programm, das eine einzugebende ganze Zahl als Bitkette ausgibt. Formulieren Sie dazu eine Funktion mit folgendem Prototyp:

```
char* crtBitStr(unsigned int x);
```

Dialog:

```
Eingabe einer int-Zahl:12000
***
dez: 12000
hex: 00002EE0
bin: 00000000|00000000|00101110|11100000
+++
```

Das Programm soll die Verarbeitungsbreite (32 bit, 64 bit) selbstständig adaptieren und dementsprechend viele Bits ausgeben. Die Größe des benötigten Puffers, in dem die Ausgabe zusammengestellt wird, soll selbstverständlich auch bytengenau ausgerechnet und in der Funktion crtBitStr definiert werden. Dazu verwende man den Operator sizeof!

Hilfestellung zur Lösung:

Mit einer Maske, die eine Eins enthält kann man ein Bit mit der bitweisen and-Operation (&) selektieren. Ist das Ergebnis Null, so war das Bit Null, ist das Ergebnis ungleich Null, so war das Bit Eins.

Mit Hilfe der Schiebeoperatoren << oder >> kann man die Zahl oder die Maske in die gewünschte Richtung verschieben.  $z = z << 1$ ; verschiebt die Zahl z um ein Bit nach links,  $z = z >> 1$ ; Verschiebt die Zahl z um ein Bit nach rechts ( $z << = 1$  oder  $z >> = 1$  ist auch möglich). Erarbeiten Sie zunächst eine Lösung ohne die Trennstriche.

---

Zusatzaufgabe:

Programmieren Sie je eine Funktion zur Ausgabekonvertierung einer int-Zahl in eine Zeichenkette (hexadezimal/dezimal). Der Speicherbereich, in dem die Zeichenkette aufgebaut werden soll, sei der erste Parameter. Der zweite Parameter enthalte die auszugebende Zahl, der 3. Parameter gebe die maximale Länge der zu erzeugenden Zeichenkette an, wobei die Zahl rechtsbündig eingetragen werden soll. Ein 4. Parameter übergibt das Füllzeichen.

Es sollen zwei Funktionen programmiert werden, eine Funktion erzeuge die dezimale, die andere die hexadezimale Zahlendarstellung. Ergänzen Sie eine main-Funktion zum Test Ihrer Lösungen.

```
/* Prototypen */
```

```
char * itoadec (char * buf, int i, int len, char leading);
char * itoahex (char * buf, unsigned int i, int len, char leading);
```

```
./a.out 12345 65535 65536
00065536 00010000
00065535 0000FFFF
```

---

[A. Beck](#)