

- Entspricht einer **Division** von i durch 2^n

`60 >> 1 == 30, 60 >> 2 == 15, 60 >> 3 == 7...`

- Siehe Links-Shift: undefiniert für $n < 0$ (Shiften um eine negative Anzahl Bits) und für $n \geq$ Bitbreite des Typs von AusdruckL (nach Konvertierung)**
- Beispiele: Bit auf bestimmte Position shiften**

```
unsigned char uc = 0;
signed char   sc = 0;
int           sint = 0;
```

```
uc = 128 >> 0;    /* 10000000 (128) */
uc = 128 >> 2;    /* 00100000 (32) */
uc = 128 >> 12;   /* 00000000 (0) */
```

```
sint = 128 >> 2;   /* 00000000 00000000 00000000 00100000 (32) */
sint = -128 >> 0;  /* 11111111 11111111 11111111 10000000 (-128) */
sint = -128 >> 2;  /* 11111111 11111111 11111111 11100000 (-32) */
/* oder */
/* 00111111 11111111 11111111 11100000 (1073741792) */
uc = (177 << 1);   /* 10110001 << 1 => 01100010 */
uc >>= 4;          /* 01100010 >> 4 => 00000110 */
```

Annahmen
char 8bit
Zweierkomplement-
Darstellung