

```

D:\Technik\Informatik\Programmierung\C++\Programme\Nachhilfe\Matthias\BinäreDarstellu
Fuer DecimalToBinaer druecken Sie '1',
fuer BinaerToDecimal druecken Sie '2' !
1
Geben Sie eine Zahl ein !15

Interne Maschinendarstellung der Zahl: 15

V Exp      Mantisse
0 10000010 11100000000000000000000_

```

Eingabe:15

15 = 1111,000... = 1,111000000000000 * 2^3 → Exponent = 3 + 127 (Charakteristik) = 130 = 10000010.

→

```

V Exp      Mantisse
0 10000010 1110000 00000000 00000000
- - - - -

```

Byte[3] = 0 1000001 = V + Exponent - letzter Stelle

Byte[2] = 0 1110000 = letzte Stelle des Exponenten + 7 letzten Mantissenstellen.

Byte[1] ...

Byte[0] ...

```
// Vorzeichen
if((g & s[3]) == 0)
    cout << 0 << " ";
else
    cout << 1 << " ";

// Exponent
g = g >> 1;
for(int exp=0; exp<7;exp++)
{
    if((g & s[3]) == 0)
        cout << 0;
    else
        cout << 1;
    g = g>>1;
}
```

// muss jetzt letztes Bit von byte[2] heranziehen, da die letzte Stelle meine
// letzte Exponentenstelle ist.

```
g = 1;
g = g << 7;
if((g & s[2]) == 0)
    cout << 0;
else
    cout << 1;
g = g>>1;
```

```
g = 1;
g = g << 6;
cout << " ";
```

```
V Exp      Mantisse
0 10000010 1110000 00000000 00000000
- - - - -

if((g & s[3]) == 0)
    cout << 0 << " ";
else
    cout << 1 << " ";

000000000...10000000 → g
& \0\0\0\0\0\0\0.... 01000001 → s[3]
- - - - -
Ergebnis: 00000000 → cout << 0 << " ";
```

```
V Exp      Mantisse
0 10000010 1110000 00000000 00000000
- - - - -

byte[3]    byte[2]    byte[1]
0 1000001 0 1110000 00000000 00000000
- - - - -

g = 1;      000000000...00000001
g = g << 7; 000000000...10000000

g & s[2])   000000000...10000000 → g
& \0\0\0\0\0\0\0.... 01110000 → s[2]
```