

```

// Beispiele zu Bitoperatoren
// Operatoren: <<, >>, ~, &, ^, |

#include <iostream>
#include <iomanip>
using namespace std;

#define bit0(x) (x&1)

// Berechnung der Parität
int parity (unsigned u) {
    int p = 0;
    for ( ; u != 0; u >>= 1)
        p ^= bit0 (u);
    return p;
} //parity

int swapbits (int x, int nr1, int nr2) {
// Vertauschen der Bit an den Positionen nr1 und nr2

    int hbp = 8 * sizeof (int) - 1;
    if ((nr1 < 0) || (nr1 > hbp)
        || (nr2 < 0) || (nr2 > hbp))
        return x;

    int mask1 = (1 << nr1);
    int mask2 = (1 << nr2);

    int erg = x & ~(mask1 | mask2);
    if (x & mask1) erg |= mask2;
    if (x & mask2) erg |= mask1;
    return erg;
} //swapbits

```

```

int main () {

    int ia = 0xf0000000;
    cout << "0xf0000000 um 4 Positionen verschoben: "
         << (ia <<4) << endl;
    cout << endl;

    char a = 'A';
    cout << "A um 4 Positionen verschoben: "
         << (a << 4) << endl;
    cout << endl;

    cout << "Parität von 0xff = " << parity (0xff) << endl;
    cout << "Parität von 301 = " << parity (301) << endl;
    cout << endl;

    cout << "swapbits (0xf0, 0, 7) = " << setw (4)
         << hex << swapbits (0xf0, 0, 7) << endl;
    cout << "swapbits (0xf0, 32, 7) = " << setw (4)
         << hex << swapbits (0xf0, 32, 7) << endl;
    cout << "swapbits (0xf0, 0, 3) = " << setw (4)
         << hex << swapbits (0xf0, 0, 3) << endl;
    cout << "swapbits (0xf0, 10, 7) = " << setw (4)
         << hex << swapbits (0xf0, 10, 7) << endl;

    return 0;
} //main

```

**/\* Ausgabe:**

**0xf0000000 um 4 Positionen verschoben: 0**

**A um 4 Positionen verschoben: 1040**

**Parität von 0xff = 0**

**Parität von 301 = 1**

**swapbits (0xf0, 0, 7) = 71**

**swapbits (0xf0, 32, 7) = f0**

**swapbits (0xf0, 0, 3) = f0**

**swapbits (0xf0, 10, 7) = 470**

**\*/**