









1 << 0 == 1 0001🡪0001

1 << 1 == 2 0001🡪0010

1 << 2 == 4 0001🡪0100

1 << 3 == 8 0001🡪1000

**Bit Setting in C**

<https://gist.github.com/Vitorbnc/e35f1ff1485d660edf365241dacfa387>

#### **Setting a bit**

Use the bitwise OR operator ( | ) to set a bit.

number |= 1 << x;

That will set bit x.

**Clearing a bit**

Use the bitwise AND operator (&) to clear a bit.

number &= ~(1 << x);

That will clear bit x. You must invert the bit string with the bitwise NOT operator (~), then AND it.

#### **Toggling a bit**

The XOR operator (^) can be used to toggle a bit.

number ^= 1 << x;

That will toggle bit x.

**Checking a bit**

To check a bit, shift the number x to the right, then bitwise AND it:

bit = (number >> x) & 1;

That will put the value of bit x into the variable bit.

**Changing the nth bit to x**

Setting the nth bit to either 1 or 0 can be achieved with the following:

number ^= (-x ^ number) & (1 << n);

Bit n will be set if x is 1, and cleared if x is 0.