didelers / bitwise-operators.md

Created Apr 12, 2012

Bitwise tricks

Inspired by this article. Neat tricks for speeding up integer computations.

Note: `cin.sync_with_stdio(false);` disables synchronous IO and gives you a performance boost. If used, you should only use `cin` for reading input (don't use both `cin` and `scanf` when sync is disabled, for example) or you will get unexpected results.

Multiply by a power of 2

```
x = x << 1; // x = x * 2
x = x << 6; // x = x * 64
```

Divide by a power of 2

```
x = x >> 1; // x = x / 2
x = x >> 3; // x = x / 8
```

Swap integers without a temporary variable

```
 a ^= b; // int temp = b
b ^= a; // b = a
a ^= b; // a = temp
```

Increment / Decrement (slower but good for obfuscating)

```
 i = -~i; // i++
i = ~-i; // i--
```

Sign flipping

```
 i = ~i + 1; // or
i = (i ^ -1) + 1; // i = -i
```

Modulo operation if divisor is power of 2

```
 x = 131 & (4 - 1); // x = 131 % 4
```

Check if an integer is even or odd

```
 (i & 1) == 0; // (i % 2) == 0
```

Equality check

```
(a^b) == 0; // a == b
```

Absolute value

```
x < 0 ? -x : x; // abs(x)
(x ^ (x >> 31)) - (x >> 31) // abs(x)
```

Equal sign check (both ints are pos or neg)

```
a ^ b >= 0; // a * b > 0
```

Rounding, ceiling, flooring

```
(x + 0.5) >> 0; // round(x)
(x + 1) >> 0; // ceil(x)
x >> 0; // floor(x)
```