Practice Problem 2.15 (solution page 184)

Using only bit-level and logical operations, write a C expression that is equivalent to **x == y**. In other words, it will return 1 when **x** and **y** are equal and 0 otherwise.

`!(x & (~y))` Only works for $y \neq 0$ (always true if $y = 0$)

`!(~x & y)` Only works for $x \neq 0$ (always true if $x = 0$)

So need to combine them,

`![(x & (~y)) | (~x & y)] = !(x ^ y)`

	0	1
0	0	1
1	1	0

↑ nonzero if x and y differ at any bit