Practice Problem 2.8 (solution page 181)

Fill in the following table showing the results of evaluating Boolean operations on bit vectors.

a	0	1	0	0	1	1	1	0
b	1	1	1	0	0	0	0	1
~a	1	0	1	1	0	0	0	1
~b	0	0	0	1	1	1	1	0
a&b	0	1	0	0	0	0	0	0
a|b	1	1	1	0	1	1	1	1
a^b	1	0	1	0	1	1	1	1

Web Aside DATA:BOOL More on Boolean algebra and Boolean rings

The Boolean operations |, &, and ~ operating on bit vectors of length w form a *Boolean algebra*, for any integer $w > 0$. The simplest is the case where $w = 1$ and there are just two elements, but for the more general case there are 2^w bit vectors of length w. Boolean algebra has many of the same properties as arithmetic over integers. For example, just as multiplication distributes over addition, written $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$, Boolean operation & distributes over |, written $a \,\&\, (b \mid c) = (a \,\&\, b) \mid (a \,\&\, c)$. In addition, however. Boolean operation | distributes over &, and so we can write $a \mid (b \,\&\, c) = (a \mid b) \,\&\, (a \mid c)$, whereas we cannot say that $a + (b \cdot c) = (a + b) \cdot (a + c)$ holds for all integers.

When we consider operations ^, &, and ~ operating on bit vectors of length w, we get a different mathematical form, known as a *Boolean ring*. Boolean rings have many properties in common with integer arithmetic. For example, one property of integer arithmetic is that every value x has an *additive inverse* $-x$, such that $x + -x = 0$. A similar property holds for Boolean rings, where ^ is the "addition" operation, but in this case each element is its own additive inverse. That is, $a \mathbin{\char`^} a = 0$ for any value $a$, where we use 0 here to represent a bit vector of all zeros. We can see this holds for single bits, since $0 \mathbin{\char`^} 0 = 1 \mathbin{\char`^} 1 = 0$, and it extends to bit vectors as well. This property holds even when we rearrange terms and combine them in a different order, and so $(a \mathbin{\char`^} b) \mathbin{\char`^} a = b$. This property leads to some interesting results and clever tricks, as we will explore in Problem 2.10.

Operation	Result
a	[01001110]
b	[11100001]
~a	
~b	
a & b	
a | b	
a ^ b	