Practice Problem 2.10 (solution page 182)

As an application of the property that $a \wedge a = 0$ for any bit vector a, consider the following program:

```
1  void inplace_swap(int *x, int *y) {
2      *y = *x ^ *y;  /* Step 1 */
3      *x = *x ^ *y;  /* Step 2 */
4      *y = *x ^ *y;  /* Step 3 */
5  }
```

As the name implies, we claim that the effect of this procedure is to swap the values stored at the locations denoted by pointer variables **x** and **y**. Note that unlike the usual technique for swapping two values, we do not need a third location to temporarily store one value while we are moving the other. There is no performance advantage to this way of swapping; it is merely an intellectual amusement.

Starting with values a and b in the locations pointed to by **x** and **y**, respectively, fill in the table that follows, giving the values stored at the two locations after each step of the procedure. Use the properties of ^ to show that the desired effect is achieved. Recall that every element is its own additive inverse (that is, $a \wedge a = 0$).

Step	*x	*y
Initially	a	b
Step 1	a	$a \wedge b$
Step 2	b	$a \wedge b$
Step 3	b	a

Step 1. $*y = *x \wedge *y = a \wedge b$

Step 2. $*x = *x \wedge *y = a \wedge (a \wedge b)$
$= (a \wedge a) \wedge b$
$= 0 \wedge b$
$= b$

Step 3. $*y = *x \wedge *y = b \wedge (a \wedge b)$
$= a \wedge (b \wedge b)$
$= a \wedge 0$
$= a$