PLAYBOOK NOTES: BIT MANIPULATION – PART 3

Now we are going to examine shift left (<<) and shift right (>>).

The shift operators are used to move bits left and right. It is worth noting that if a bit falls off the end of a variable, the bit is lost and does not wrap around to the other side. We will show this later in this document.

Let's start with the example from Part 1 where we checked to see if the 5th bit is equal to a 1 or a 0.

In part 1, the answer was 0 because the 5th bit of the code was 0.

For this example, let's say the 5th bit of code was a 1 and our answer is now equal to MASK_5.

The following is a picture of MASK_5.

BIT:	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
MASK_5:	0	0	1	0	0	0	0	0

If we want to move the 5th bit to bit 0, we could use the following code:

```
// Determine if the 5th bit is a 1
answer = code & MASK_5;

// Move the 5th bit to the right 5 spots
answer = answer >> 5;
```

If the 5th bit was a 1, answer would equal 1. If the 5th bit was a 0, answer would be 0 (actually, it would have been 0 before the shift, but that is not important 😊).

Knowing what we know now, we could write the if-statement like this:

```
// Check to see if bit 5 is a 1
if ((code & MASK_5) >> 5 == 1)
{
```

We also know that any integer that is NOT equal to 0 is true, so we could shorten the if statement to:

```
// Check to see if bit 5 is a 1
if ((code & MASK_5) >> 5)
{
```

We can do the same type of work with shift left (<<). Since the effect is just the reverse, (bits shift to the left), we will not show every detail of it. But it is important to remember that there our 4 bytes (32 bits) to every integer. So, if you shift to the left, those other 3 bytes become trivial.

Let's look at one last example of how bits can be lost.

Given the following number:

BIT:	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Code:	1	0	0	0	1	1	1	1

If we shift this to the right 5 spots (code = code >> 5), we will get the following result:

BIT:	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Code:	0	0	0	0	0	1	0	0

The 7th bit is moved to the 2rd bit and all the other bits fell off.