

## CSCI 2330 – Bit Puzzle Exercises

Assuming that **X** and **Y** are 32-bit, 2's complement (signed) ints, write an expression using **X** and/or **Y** that evaluates to each value specified below.

Operators are restricted to the following: bitwise AND, OR, NOT, XOR, left shift, arithmetic right shift, addition, and logical NOT, i.e.:

**& | ~ ^ << >> + !**

Constants are restricted to values in the range 0-255 (0x00 through 0xFF).

1. A value that is the negation of X.

Example: 0000.....0000011  $\Rightarrow$  1111....111111101 (i.e., 3  $\Rightarrow$  -3)

(Hint: 2's complement)

2. A value that contains the 6th bit of X (counting from LSB) with all other bits zeroed.

Examples: 1111.....111111111  $\Rightarrow$  0000.....000100000

0000.....000000000  $\Rightarrow$  0000.....000000000

0000.....010000001  $\Rightarrow$  0000.....000000000

(Hint: use a mask)

3. The value 1 if X equals Y and the value 0 otherwise.

(Hint: use XOR)

4. The value 1 if the 15th bit of X (counting from LSB) is a 1 and the value 0 otherwise.

(Hint: remember the limits on constant sizes)

5. The value -1 if X < 0 and the value 0 otherwise.

(Hint: sign extension)

6. Either the value -1 if X < 0, the value 0 if X equals 0, or the value 1 if X > 0.

(Hint: start with prior expression and augment with OR)