CMSC 216 Bitwise Operations Worksheet

1. Convert 3455 to binary using the division approach where you keep dividing by 2.
2. Convert 010101011 to decimal.
3. Convert 3455 to hexadecimal and then to binary.
4. Find the 2's complete represention of -17 and -19 assumming integers with 8 bits.
5. Write a function that prints the bits of an unsigned integer.
6. Write a function that rotates the bytes (not bits) of an unsigned integer to the right by 1 byte (the rightmost byte will become the leftmost one). You may not use any loop constructs.
7. Implement the function unsigned int **reverse_bytes(unsigned int x)** that reverses the bytes of x (you may assume ints are 4 bytes). For example, reverse_bytes(0x12345678) = 0x78563412. The only allowed operators are `+ - = & | ~ ^ << ! >>` and the only allowed constants are 1 2 4 8 16.
8. Implement the function **int bit_or(int a, int b)** that returns the bitwise OR of the and and b parameter (a — b). The only allowed operators are `+ & ~ ^ << ! >>`.