1. Translate the following hexadecimal numbers to binary.
   0x0, 0x10, 0xF, 0x1F, 0xA4, 0xFF

2. Find the bitwise and, or and xor of the following:
   a) 0xC6 with 0x35
   b) 0x19 with 0x24
   c) 0xD3 with 0xC7
   d) 0x17 with 0xFF

3. Find the 1's complement of the following:
   0xC6, 0x35, 0xD3 and 0xC7.

4. In this question & is bitwise and, | is bitwise or, ^ is bitwise xor, and ! is 1's complement. a is any given two digit hexadecimal number. Explain why each of the following identities holds.

   a) 0xFF & a = a. (0xFF is the identity for AND)
   b) 0xFF | a = 0xFF. (0xFF is the absorbent for OR)
   c) 0xFF ^ a = !a
   d) 0 & a = 0. (0 is the absorbent for AND)
   e) 0 | a = a. (0 is the identity for OR)
   f) 0 ^ a = a. (0 is the identity for XOR)
   g) a ^ a = 0 (a is its own inverse under XOR)
   h) For any three two digit hexadecimal numbers a, b and c:
      If a ^ b = c then a ^ c = b.

5. Write a program that counts the number of bits set in an integer. For example, the number 5 (decimal), which is 0000000000000101 (binary), has two bits set.

6. Write a program that takes a 32-bit integer (long int) and splits it into eight 4-bit values. (Be careful of the sign bit.)

7. Write a program that will take the bits in a number and shift them to the left end. For example, 01010110 (binary) would become 11110000 (binary).