Masterclass in C - Exercise Set 003

This exercise is related to format conversions of integers across four number bases, viz. binary, octal, decimal, and hexadecimal. You will be given a set of numbers in each of the above number bases, and you must convert the same into the rest of the bases. For example, given a decimal number must be converted into binary, octal and hexadecimal, given an octal number must be converted into binary, decimal and hexadecimal and so forth.

1. Convert the following decimal numbers into binary, octal and hexadecimal.
    a) 100
    b) 200
    c) 7387
    d) 1192
    e) 124678
    f) 2534112
2. Convert the following binary numbers into octal, decimal, and hexadecimal.
    a) 110011100010101001
    b) 101011010101111001101101
    c) 10100000111111000111100000101
    d) 1100110011001100
    e) 1010
    f) 10101010
    g) 11111111
    h) 1111111111111111
    i) 11111111111111111111111111111111
3. Convert the following hexadecimal numbers into binary, octal and decimal.
    a) 12AB45C
    b) FFFF
    c) FFFFFFFF
    d) A0B0C0D0
    e) 1982
    f) BFFF
    g) 3FFFFFFF
    h) 40000000
    i) 7FFFFFFF
    j) 80000000
    k) BFFFFFFF

l) C0000000
m) 100000000
n) ABCDEF

4. Covert the following octal numbers into binary, decimal and hexadecimal.
    a) 777
    b) 14642
    c) 762653
    d) 77777
    e) 1717
    f) 2626
    g) 41723

5. Convert the following numbers into binary and store them to fit into a given number of bytes.

    Solved Problem:

    Question: Convert decimal 150 binary and store in 1 byte, 2 bytes and 4 bytes.

    Solutions:
    First, convert 150 into binary.
    $(150)_{10} = (10010110)_2$

    If the number of bits is between 1 to 8, then min. 1 byte is required.
    If the number of bits is between 9 to 16, then min 2 bytes are required.
    If the number of bits is between 17 to 32, then min 4 bytes are required.
    [Remember we do not have 3 bytes integer]

    In the above example, 150 fits in 1 byte. Therefore 150 decimal is stored in 1 byte.
    [10010110]
    Byte 1

    If 150 fit in two bytes, then keep all bits in byte 2 to zero.
    [00000000] [10010110]
    Byte 2 Byte 1

    If 150 were to fit in four bytes, then keep all bits in bytes 2, 3 and 4 to zero.
    [00000000] [00000000] [00000000] [10010110]
    Byte 4 Byte 3 Byte 2 Byte 1

Based on the solved problem, answer the following:

a) Store decimal 255 in 1 byte, 2 bytes and 4 bytes
b) Store decimal 62189 in 2 bytes and 4 bytes
c) Store hexadecimal FA21 in 2 bytes and 4 bytes
d) Store hexadecimal FBC234 in 4 bytes
e) Store octal 71234 in the required number of bytes
f) Store decimal 36476325 in the required number of bytes
g) Store hexadecimal A0B0C0D0 in the required number of bytes.

Hints: Read only if necessary.

Hint for solving problem 1:

For converting a decimal number to binary, divide the number repeatedly by 2, registering the quotient and remainder for each division. Follow this process until the quotient drops down to zero. Arrange all remainders in the reverse order when the quotient drops to zero. For converting the decimal number of octal and hexadecimal, follow the same process but divide by 8 and 16, respectively, instead of 2.

Hint for solving problem 2:

Convert each binary number into decimals by the method explained in class (positional number system). Thus, binary to decimal is done. And then convert the decimal into octal and hexadecimal as you have done in problem 1.

Binary → Decimal and then Decimal → Octal and Decimal → Hexadecimal.

Hint for solving problem 3:

Convert hexadecimal to decimal first by using a positional number system. And then convert decimal into binary and octal as you have done in problem 1.

Hexadecimal → Decimal and then Decimal → Binary and Decimal → Octal.

Hint for solving problem 4:

Octal → Decimal and then Decimal → Binary and Decimal → Hexadecimal.