Homework 4: Number Base Conversions

CS 200 • 10 Points Total
Due Wednesday, March 1, 2017

Assignment

Perform the following number base conversions. All necessary steps to convert by hand should be shown - any conversion without accompanying work will be counted wrong. (1 pt each)

1. Convert 51360_7 to base 10.

   $5 \times 7^4 + 1 \times 7^3 + 3 \times 7^2 + 6 \times 7^1 + 0 \times 7^0 = 12005 + 343 + 147 + 42 + 0 = 12537$

2. Convert 3157_{10} to base 16.

   Easy way: 256 < 3157 < 4096, so 256 is the highest power of 16 needed. $3157 \div 256 = 12\ r85 \rightarrow 85 \div 16 = 5\ r5$. So the digits are 12, 5, and 5. But 12 is 'C' in base 16, so 439_{10} = $C55_{16}$

3. Convert 173_{10} to base 2.

   Easy way: 128 < 173 < 256, so 128 is the highest power of 2 needed. Write down the places: 128 64 32 16 8 4 2 1. 173 – 128 = 45, so the remainder uses 32. 45 – 32 = 13, so the remainder uses 8. 13 – 8 = 5, so we use 4. 5 – 4 = 1 which means we use the 1s place and we are done. Now we put a 1 in every place we used and a 0 in every place we skipped.

128	64	32	16	8	4	2	1
1	0	1	0	1	1	0	1

   $173_{10} = 10101101_2$

4. Convert $D1B079_{16}$ to base 2.

   Each hex digit is four binary digits. D = 1101, 1 = 0001, B = 1011, 0 = 0000, 7 = 0111, and 9 = 1001. String them together to get the binary number:

   $D1B079_{16} = 110100011011000001111001_2$

5. Convert 101010111001_2 (unsigned) to base 10.

   $$1 \times 2^{11} + 0 \times 2^{10} + 1 \times 2^9 + 0 \times 2^8 + 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$
   $$= 2048 + 0 + 512 + 0 + 128 + 0 + 32 + 16 + 8 + 0 + 0 + 1 = 2745_{10}$$

6. Convert 101010111001_2 (1s Complement) to base 10.

   If the leftmost bit were a 0, it would be a positive number and we could simply do the same as the previous problem. But the leftmost bit is a 1, so it is negative and we must first flip all the bits to get the positive complement: 010101000110_2. Now we proceed:

$$0 \times 2^{11} + 1 \times 2^{10} + 0 \times 2^9 + 1 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$
$$= 0 + 1024 + 0 + 256 + 0 + 64 + 0 + 0 + 0 + 4 + 2 + 0 = 1350_{10}$$

   But don't forget that it's negative, so the final answer is -1350_{10}

7. Convert 101010111001_2 (2s Complement) to base 10.

   Similarly to the last problem, a 0 in the leftmost bit would make the problem simple. But to reverse the process for a negative, we first flip the bits to get the 1s complement: 010101000110_2. Next we add 1 to get the 2s complement: 010101000111_2. Now we proceed:

$$0 \times 2^{11} + 1 \times 2^{10} + 0 \times 2^9 + 1 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$
$$= 0 + 1024 + 0 + 256 + 0 + 64 + 0 + 0 + 0 + 4 + 2 + 1 = 1351_{10}$$

   Again, don't forget the negative sign: -1351_{10}

8. Convert 320147_8 to base 16.

   Base 8 converts to 3-bit binary numbers: 3 = 011, 2 = 010, 0 = 000, 1 = 001, 4 = 100 and 7 = 111. Now we put them all together: 011010000001100111. Then, starting at the right, divide them into groups of four: 01 1010 0000 0110 0111. Finally, convert the 4-bit binaries to hex digits: 0001 = 1, 1010 = A, 0000 = 0, 0110 = 6, and 0111 = 7. 320147_8 = $1A067_{16}$

9. Convert $5CF9_{16}$ to base 10.

$$5 \times 16^3 + 12 \times 16^2 + 15 \times 16^1 + 9 \times 16^0 = 20480 + 3072 + 240 + 9 = 23801_{10}$$

10. Convert -730_{10} to a **<u>12-bit</u>** 2s Complement.

    512 is the highest power of 2 needed for 730. Write the powers down:

512	256	128	64	32	16	8	4	2	1
1	0	1	1	0	1	1	0	1	0

    That's only 10 bits, so pad 0s on the left and flip the bits: 110100100101. Then add 1 to get the final answer: 110100100101 + 1 = 110100100110_2.