ACSL: Computer Number Systems

Sanjit Bhat Alexander Sun

October 5, 2018

1 Easy to Medium Problems

1. Convert 8765_{16} to binary form.
2. Evaluate in base 16, $\text{FEED}_{16} - \text{6ACE}_{16}$
3. What is the base 16 representation for $\text{FEDCBA}_{16} - \text{ABCDEF}_{16}$?
4. What is $\frac{\text{A98}_{16}}{23_8}$ in base 10 (leave it in simplified fraction form if necessary)?
5. In the ACSL computuer, each "word" of memory contains 20 bits representing 3 pieces of information. The most significant 6 bits represent Field A; the next 11 bits, Field B; and the last 3 bits represent Field C. For example, the 20 bits comprising the "word" 18149_{16} has fields with values of 6_{16}, 29_{16}, and 1_{16}. What is Field B in E1B7D_{16}? (Express your answer as a base 16 number).
6. $X37_8 = 1X\text{F}_{16}$. Find an X that satisfies this equality.
7. Which of the following 5 numbers is the largest? F1_{16}, 375_8, 10F_{16}, 264_{10}, or 11111000_2.

1.1 Solutions

1. Split each base 16 digit into its 4-digit binary equivalent.

$$8765_{16} = \boxed{1000\ 0111\ 0110\ 0101_2}$$

2. First, we convert to base 10.

$$\text{FEED}_{16} = (15 * 16^3 + 14 * 16^2 + 14 * 16 + 13)_{10} = 65261_{10}$$
$$\text{6ACE}_{16} = (6 * 16^3 + 10 * 16^2 + 12 * 16 + 15)_{10} = 27343_{10}$$

$65261_{10} - 27343_{10} = 37918_{10}$. Just as a reminder, we'll do the last bit of computation to convert to base 16.

$$37918/16 = 2369 \text{ R } 15$$
$$2369/16 = 148 \text{ R } 1$$
$$148/16 = 9 \text{ R } 4$$
$$9/16 = 0 \text{ R } 9$$

Put together the remainders in reverse order of when we got them, and our final number is $\boxed{941\text{E}_{16}}$

3. Look at Problem 2 for a full explanation. Answer is $\boxed{530\text{ECB}_{16}}$

4. $$\frac{\text{A98}_{16}}{23_8} = \boxed{\frac{2712_{10}}{19_{10}}}$$

5. Just some definitions first. Most significant bit refers to the bit that changes least as the number gets larger (typically the bit farthest to the left). Conversely, least significant bit refers to the bit that often changes as the number changes (typically the bit farthest to the right).

$$\begin{aligned}\text{E1B7D}_{16} &= 1110\ 0001\ 1011\ 0111\ 1101_2 \\ &= 111000\ 01101101111\ 101_2\end{aligned}$$

Therefore, $\text{A} = 38_{16}$, $\boxed{\text{B} = 36\text{F}_{16}}$, and $\text{C} = 5_{16}$.

6. $$X37_8 = 1X\text{F}_{16}$$
$$(X * 8^2 + 3 * 8 + 7)_{10} = (1 * 16^2 + X * 16 + 15)_{10}$$
$$(64X + 24 + 7)_{10} = (256 + 16X + 15)_{10}$$
$$(48X)_{10} = 240_{10}$$
$$\boxed{X_{10} = 5}$$

7. You can technically convert to whatever base you want and compare. However, I'd suggest you convert to base 2 since 4 out of 5 of the numbers given are in a power of 2 base (and thus can be easily converted to base 2). The answer is $\boxed{10\text{F}_{16}}$, which is 100001111 in base 2.

2 Hard Problems

1. How many numbers from 300 to 500, inclusive, have 8 1's in their binary representation?

2. Let n be any positive base 10 integer from 1 to 2^{12} inclusive. Let $S(n)$ be the number of 1's in the binary representation of n. Find the number of possible n's such that $S(n) - S(n+1) = 3$.

3. Find the number of carries when summing 2345_{10} and 3459_{10} (Check out Kummer's Theorem).

2.1 Solutions

1. The first step is to find the smallest number that has 8 1's in its binary representation. Clearly, that number is 11111111_2.

   Next, we'll use a neat little formula (which we'd highly recommend that you memorize) to convert special binary numbers to decimal. This formula states that a binary number with only d digits all set to 1 will have a decimal representation of $2^d - 1$. Proof of the formula follows from noticing that the next binary number (a 1 with d 0's behind it) is precisely 2^d.

   $$11111111_2 = 2^8_{10} - 1_{10} = 256_{10} - 1_{10} = 255_{10}$$

   Now that we know the smallest number which satisfies the constraint is 255, we would like to work our way up to find another such number. The purpose of doing this is to find the start of a sequence of binary numbers which could possibly be within the 300-500 range.

   The binary numbers larger than 11111111_2 all have more than 9 or more digits, and the first 9 digit binary number to have after We can see that we have to add a 0 to the binary representation, which always results in a 9 digit binary number leading with 1.(We can't place a 0 at the beginning). Therefore the next smallest number complying with the rule that we can make is 101111111

   The value of 2^8 by itself is 512 and already is above the 500 limit. So just by attempting to make the next smallest number with 8 1's in its binary representation we rise over the 500 limit. So the answer is $\boxed{0}$.

   This last paragraph and answer is wrong LOL. Let's get to the correct answer in the meeting.

2. First of all, we see that adding 1 to the binary representation of a number, decreases it's 1 count by 3.

   Even numbers in binary end with a 0 in the last place and odd binary numbers end with a 1. When 1 is added to an even binary number the amount of 1's increases by one: $110 + 1 = 111$. The one added never carries over to the other digits in the binary representation

Odd numbers in binary end with a 1 in the last place. When 1 is added it causes the last digit to change from a 1 to a 0 and carry over to the next place. $101 + 1 = 110$. The carry over is the key as when we have a chain of 0's tailing the number we end up with a chain of transformations: $1011 + 1 = 1100$. As you can see the chain of 2 1's was replaced a single 1 that carried over into the next place.

The next important pattern that we see is that the chain is always replaced by a singular 1 in the next place. Now looking back at the problem, it asks us to find numbers with a loss of 3 1's when 1 is added to it's binary representation. Because the chain always carries a 1 into the next place, we need a chain of 4 1's at the tail to create a loss of 3. We conclude that our tail must always end with 01111.

Now we must figure out how to deal with the limit of 2^{12}. We know that a binary number with all ones with 12 digits is equivalent to $2^{12} - 1$ Even attempting to place one 1 in the 1st place of a 13 digit number with the rest of values being 0 is out of range. We conclude that the largest number that satisfies this condition is 111111101111. Now to finally solve the problem each digit place after the set tail has 2 options. Either a 1 or 0 and all combinations fit the solution because at most we have that number above. Because there are 7 digits that we can modify this way, the solution is 2^7

3. Check out https://planetmath.org/kummerstheorem.

A Intuition for Base 10 to Base b Conversion Process

Recall in Problem 1.2 our process for converting 37918_{10} into base 16.

$$\begin{aligned} 37918/16 &= 2369 \text{ R } 15 \\ 2369/16 &= 148 \text{ R } 1 \\ 148/16 &= 9 \text{ R } 4 \\ 9/16 &= 0 \text{ R } 9 \end{aligned}$$

In this section, we'd like to understand why putting together the remainders in reverse order yields our final number, 941E_{16}.

To show this, first, let's re-write the above the first step as

$$37918 = 2369 * 16 + 15$$

or, more generally, as

$$n_1 = q_1 * b + r_1 \tag{1}$$

where q_1 is the quotient and r_1 is the remainder when the original number n_1 is divided by the base b.

Notice that r_1 is always less than b, so it fits perfectly that r_1 is also the first digit from the right of the base b representation of n_1. One down, the rest to go!

The key to finding the other digits is to notice that Equation 1 is highly similar to the definition of a base b number used earlier. In fact, q_1 is exactly the base 10 representation

of the base b number without its left-most digit. It's a recursive problem! As such, we can apply the same factorization solution to each quotient, take the remainders, and extract out every single digit! Since each factorization brings another multiplication by b, remainders computed earlier appear as digits farther to the left than remainders computed later. Hence, the reverse order.