CS 7A – Spring '13 Lab 02 – Show your work for credit Name________________________

1. Determine what 4 one-byte integers are stored in these 4 bytes:
2. Determine what 2 two-byte integers are stored in these 4 bytes.
3. Determine what four-byte integer is stored in these 4 bytes.
4. Interpret the 4 bytes as 4 ASCII characters.
5. Write 6666 in binary form.
6. Write 6666 in Hex form.
7. Convert the hex number fafa to decimal form.

⋮
00101010
01101101
10001001
10001111
⋮

8. Write the polynomial $p(x) = 8x^4 + 4x^3 + 2x^2 - 1$ in Horner's form and use that form to evaluate $p(1.2)$, step by step.

9. How many multiplications does Horner's method use for this polynomial vs. the method of directly evaluating the polynomial term by term?

10. Write a C++ program to solve the equation $ax + b = c$ where the user supplies *a*, *b* and *c* and the program returns the solution, if there is one, or an error message otherwise. Submit the program using <your initials>lab2.cpp to ghagopian@collegeofthedesert.edu

SOLUTIONS:

⋮
00101010
01101101
10001001
10001111
⋮

1. Determine what 4 one-byte integers are stored in these 4 bytes:
   $00101010_2 = 2^5+2^3+2^1 = 32+8+2 = 42_{10}$
   $01101101_2 = 2^6+2^5+2^3+2^2+2^0 = 64+32+8+4+1 = 109_{10}$
   $10001001_2 = 2^7+2^3+2^0 = 128+8+1 = 137_{10}$
   $10001111_2 = 2^7+2^3+2^2+2^1+2^0 = 128+8+4+2+1 = 143_{10}$
2. Determine what 2 two-byte integers are stored in these 4 bytes.
   ANS: This depends on which is the more significant byte.
   Thus the first two bytes could be interpreted as either
   $0010101001101101_2 = 42*2^8 + 109 = 10861$ or
   $0110110100101010_2 = 109*2^8 + 42 = 27946$
   and the second two bytes could be
   $1000100110001111_2 = 137*2^8 + 143 = 35215$ or
   $1000111110001001_2 = 143*2^8 + 137 = 36745$
3. Determine what four-byte integer is stored in these 4 bytes.
   ANS: Again there are two reasonable interpretations, depending on whether the first or last byte is the most significant:
   $00101010011011011000100110001111_2 = 10861*2^{16} + 35215 = 711821711$ or
   $10001111100010010110110100101010_2 = 36745*2^{16} + 27946 = 2408148266$

4. Interpret the 4 bytes as 4 ASCII characters.
ANS: The original ASCII character set had only 128 characters, and the printable characters were from 32–126. Going to the standard extended ASCII table or at the site http://en.wikipedia.org/wiki/Code_page_437 we can look up the values. Note that, along with the characters in the range 0 to 31 (00_{hex} to $1F_{hex}$), which can be interpreted as ASCII controls as well as graphical "dingbats," some characters have overloaded meanings.
Executing the following code snippet gives a good idea of how our C++ compiler interprets these:

```
cout << char(42) << endl; //asterisk
cout << char(109) << endl; // m
cout << char(137) << endl; // (e-umlaut or diaeresis)
cout << char(143) << endl; // A with a little circle above
```

we see that

```
char(42)=*, char(109)=m, char(137)=ë and char(143)=Å
```

5. Write 6666 in binary form.

ANS: The simplest algorithm for converting from decimal to binary is Horner's algorithm. To illustrate how this works, consider the number
$37 = 32 + 4 + 1 = 1*2^5 + 0*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 1*2^0$.

Horner's form for this is $(((((0*2+1)*2 + 0)*2 + 0)*2 + 1)*2 + 0)*2 + 1$.
So we note that the number is odd by putting down a "1", subtracting one and dividing by 2.

Now we have
$18 = 16 + 2 = 1*2^4 + 0*2^3 + 0*2^2 + 1*2^1 + 0*2^0 = ((((0*2+1)*2 + 0)*2 + 0)*2 + 1)*2 + 0$.

We note that this is even so put down a zero, for "01" and divide by 2:
$9 = 8 + 1 = 1*2^3 + 0*2^2 + 0*2^1 + 1*2^0 = (((0*2+1)*2 + 0)*2 + 0)*2 + 1$

This is odd so put down a one for "101" and repeat division by 2:
$4 = 1*2^2 + 0*2^1 + 0*2^0 = ((0*2+1)*2 + 0)*2 + 0$

Again it's even so we have "0101" and dividing by 2:
$2 = 1*2^1 + 0*2^0 = (0*2+1)*2 + 0$

Even again so we get "00101" and divide by 2:
$1 = 1*2^0 = 0*2+1$

And then finally, "100101" is our binary number.

k	***x***	b_k
0	6666	0
1	3333	1
2	1666	0
3	833	1
4	416	0
5	208	0
6	104	0
7	52	0
8	26	0
9	13	1
10	6	0
11	3	1
12	1	1
	0	

Applying this algorithm to 6666, the results are tabulated at right, showing that $6666 = 1101001100010_2 = 2^{12}+2^{11}+2^9+2^6+2^5+2$
or 4096 + 2048 + 512 + 8 + 2

6. Write 6666 in Hex form.
   ANS: Use algorithm 1.5 from the text: Decimal Integer to Hexadecimal
   To convert the integer ***x*** into its equivalent hexadecimal numeral:
   1. Assert ***x*** > 0.
   2. Set ***x*** = 0.
   3. Divide ***x*** by 16, setting ***x*** equal to the (integer) quotient.
   4. Set h_k equal to the remainder from the previous division.
      Use one of the 16 ***hexadecimal digits*** 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f, representing the numbers 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, for h_k.
   5. Add 1 to ***k.***
   6. If ***x*** > 0, repeat steps 3-6.
   7. Return $\mathrm{h_k \cdots h_2 h_1 h_0}$ (*i.e.*, the hexadecimal numeral whose ***j***th hex symbol is h_j.)

k	***x***	Quotient after division by 16	Remainder after division by 16
0	6666	416	`10 = a`
1	416	26	0
2	26	1	`10 = a`
3	1	0	1

   This is then $1 \times 16^3 + 10 \times 16^2 + 0 \times 16^1 + 10 \times 16^0 = 4096 + 2560 + 10 = 6666$

7. Convert the hex number `fafa` to decimal form.
   ANS: Since f is hex for 15 and a is hex for 10, this is $\mathtt{1a0a}_{16}$ =
   $15 \times 16^3 + 10 \times 16^2 + 15 \times 16^1 + 10 \times 16^0 = 61440 + 2560 + 240 + 10 = 64250$

8. Write the polynomial $p(x) = 8x^4 + 4x^3 + 2x^2 - 1$ in Horner's form and use that form to evaluate $p(1.2)$, step by step.
   ANS: $p(x) = 8x^4 + 4x^3 + 2x^2 - 1 = (((8x + 4)x + 2)x + 0)x - 1$
   So $p(1.2) = (((8 \cdot 1.2 + 4) \cdot 1.2 + 2) \cdot 1.2 + 0) \cdot 1.2 - 1 = (((13.6) \cdot 1.2 + 2) \cdot 1.2 + 0) \cdot 1.2 - 1$
   $= ((18.32) \cdot 1.2 + 0) \cdot 1.2 - 1$
   $= (21.984) \cdot 1.2 - 1$
   $= 25.3808$

9. How many multiplications does Horner's method use for this polynomial vs. the method of directly evaluating the polynomial term by term?
   ANS: The $p(x) = 8 \cdot x \cdot x \cdot x \cdot x + 4 \cdot x \cdot x \cdot x + 2 \cdot x \cdot x - 1$ form requires 9 multiplications whereas $p(1.2) = (((8 \cdot 1.2 + 4) \cdot 1.2 + 2) \cdot 1.2 + 0) \cdot 1.2 - 1$ requires only 4.