CS 3843 Computer Organization, Fall 2013 Assignment 1 Solution

Due Friday, September 13, 2013

This assignment is due at the beginning of class on the due date. There will be a 10 percent penalty for late assignments.

Solve the following problems and hand them in at the beginning of class on the due date. You may use a basic calculator, but you must show how you got your answer.

Write out the solutions neatly. Problems should be in order. Your **name** and the assignment number should be in the upper right corner of the first sheet you hand in. Stack the pages neatly and put a single stable in the upper left corner. Make sure the staple does not obscure any of your writing.

1. Convert to decimal: 0x3d7e

   Ans: $(3 * 16^3) + (13 * 16^2) + (7 * 16^1) + (14 * 16^0)$

   $= (3 * 4096) + (13 * 256) + 112 + 14$

   $= 12288 + 3328 + 112 + 14$

   $= 15742$

2. Convert to binary: 0x3d7e

   Ans: 0x3d7e = [0011 1101 0111 1110]

3. Convert from decimal to binary: 764

   Ans:

   764 = (47 * 16 + 12)
   47 = (2 * 16 + 15)

   Hence 764 = 0x2FC =[0010 1111 1100]

4. Convert from decimal to hexadecimal: 764

   Ans:

   764 = (47 * 16 + 12)
   47 = (2 * 16 + 15)
   Hence 764 = 0x2FC

5. Determine the output of the following code segment (without running it):
   Assume that an int is 32 bits.

```
unsigned a = 236;
unsigned b = 163;
unsigned c = ~a;
unsigned d = a & b;
unsigned e = a | b;
```

```
unsigned f = a ^ b;
printf("%u %u %u %u %u %u\n",a,b,c,d,e,f);
```

Ans: 236 163 4294967059 160 239 79

6. Write a code segment that uses shifts to multiply the unsigned int *x* by 36.

   Ans: x * 36

   = x * (32 + 4)

   = x* (2^5 + 2^2)

   Hence (x << 5) + (x <<2)

7. Using an 8-bit word, find the binary representation of -39 in
   a. two's complement
   b. ones' complement
   c. sign-magnitude

   Ans:

   a. [11011000] + 1 = [11011001] or 256 – 39 = 217 = [11011001]

   b. [11011000]

   c. [10100111]

8. Using a 16-bit word, find the binary representation of -39 in
   a. two's complement
   b. ones' complement
   c. sign-magnitude

   Ans:

   a. [11111111 11011001]

   b. [11111111 11011000]

   c. [10000000 00100111]

9. Assume that a short is represented by 11 bits and an int is represented by 17 bits. What is the output generated by the following code segment:

```
int x = 3245;
int y = -3245;
short sx = (short)x;
short sy = (short)y;
printf("%d %d %d %d\n",x, y, (int)sx, (int)sy);
printf("%x %x %x %x\n",x, y, (int)sx, (int)sy);
printf("%u %u %u %u\n",x, y, (int)sx, (int)sy);
```

   You may use a calculator to generate the values, but you must show how you calculated them.

Ans:

x = 3245 = 0xCAD = [0 0000 1100 1010 1101]

y = -3245 = [1 1111 0011 0101 0010] + 1 = [1 1111 0011 0101 0011] = 0x1F353

sx = (short)x = [100 1010 1101] = - [011 0101 0011] = -[0x353] = -851

sy = (short)y = [011 0101 0011] = 0x353 = 851

(int)sx = [1 1111 1100 1010 1101] =0x1FCAD= - [0 0000 0011 0101 0011] = -851

(int)sy = [0 0000 0011 0101 0011] = 0x353 = 851

Output will be

3245, -3245, -851, 851

0xCAD, 0x1F353, 0x1FCAD, 0x353

3245, 127827, 130221, 851

10. Dell computers have a unique alpha-numeric service tag consisting of digits and upper case letters. When you call Dell technical support, they would like to know the service tag number so that they can appropriately route your call. One way to automate this is to have you type in your service tag number. Since there is no convenient way to uniquely type letters on a standard telephone keypad, Dell also gives you an express code, which is a decimal number calculated from the service tag. The calculation algorithm is simple: the service tag is treated as a base-36 number, where A=10, B=11, etc.

    Find the express code that corresponds to the service tag: M7D3XY1. You must show how you got your answer.

    Ans:

    M7D3XY1 = $(M * 36^6) + (7 * 36^5) + (D * 36^4) + (3 * 36^3) + (X * 36^2) + (Y * 36^1) + (1 * 36^0)$

    $= (22 * 36^6) + (7 * 36^5) + (13 * 36^4) + (3 * 36^3) + (33 * 36^2) + (34 * 36^1) + (1 * 36^0)$

    $= (22 * 2176782336) + (7 * 60466176) + (13 * 1679616) + (3 * 46656) + (33 * 1296) + (34 * 36) + 1$

    = 48334493593