Problem Set 1

Vidya Akavoor
9/22/16

Part B - Book Problems

1) A. !!x
   B. !!(~x)
   C. !!(x & 0xff)
   D. !!~(x & 0xff)

2)
```
int leftmost_one(unsigned x){
    x |= x >>16;
    x |= x >>8;
    x |= x >> 4;
    x |= x >> 2;
    x |= x >> 1;
    x ^= x >>1;
    return x;
}
```

3)
```
int lower_one_mask(int n){
    int x = 1<<(n-1);
    x = (x << 1) -1;
    return x;
}
```

4)
```
int saturating_add(int x, int y){
    int sum = x+y;
    int sum_sign = sum >>31;
    int x_sign = x >>31;
    int y-sign = y>>31;
    int pos_overflow = ((x_sign ==0)&& (y_sign==0)&& (sum_sign== 1)
    int neg_overflow = ((x_sign==1) && (y_sign ==1)&& (sum_sign == 0)
    int test1 = (pos_overflow ==1);
    int test2 = (neg_overflow ==1);
```

⇒

```
return (((test1<<31)>>31 & INT_MAX)
        |((((!test1&test2)<<31)>>31) & INT_MIN)
        |((((!test1& !test2)<<31)>>31)& sum);
}
```

5) A. ~((1<<k)-1)   B. ((1<<k)-1)<<j

2) 2's Complement Respresentation

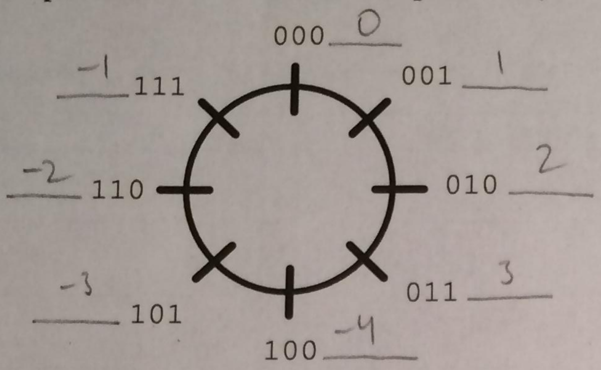
Fill in the below table assuming a 64 bit computer that uses 2's complement representation, `INT_MAX` and `INT_MIN` are defined as the computer's signed integer representation maximum and minimum value respectively, and:

```
int x = INT_MIN, y = 0xdecafbad, z = INT_MAX, i = (sizeof(char *) + sizeof(int *));
```

C Expression	Hexadecimal
x	0x 80000000
y	0x decafbad
z	0x 7FFFFFFF
i	0x 00000010
z<<3	0x FFFFFFF8
z<<((i>>1)-1)	0x FFFFFF80
~0 == (z + INT_MIN)	0x 00000000
y & 0xffff	0x 0000fbad
y >> 16	0x ffffdeca
(y >> 16) | 0xffff	0x ffffffff
(~(0x10>>2)+1) == -(i>>2)	0x 00000001
(~z+1) + -1	0x 80000000
(~((~x) << 1)) & y	0x 00000001
((y<<3)+INT_MIN)^((y<<3)+INT_MIN)	0x 00000000

3) Misc

1. Complete the following diagram – write down, on the dashed lines, the signed integer base 10 equivalents for a 3 bit 2's complement system.



2. $(101111)_2$ to base 10 47
3. $(101010)_2$ to base 16 2A
4. $(65)_{10}$ to binary 1000001
5. $(347)_8$ to base 16 E7
6. Find the decimal equivalent of the 13-bit twos complement number: 1111111111111 -1
7. Show the results of adding the following pairs of 4-bit twos complement numbers in decimal and indicate whether or not overflow occurs for each case.
   (a) 1111 + 1111 → -1 + -1 = -2 → 1110 ; no overflow
   (b) 1010 + 0111 → -6 + 7 = 1 → 0001 ; no overflow
   (c) 1110 + 0001 → -2 + 1 = -1 → 1111 ; no overflow
8. Complete the following table for the 5-bit 2's complement representation. Show your answers as signed base 10 decimal integers and the 2's complement binary value.

value	decimal	binary
Largest Positive Number	15	01111
Most Negative Number	-16	10000
Number of distinct Numbers	31	11111