

Problems 1-4

What are the decimal values of each of the following binary numbers if you interpret them as 2's complement integers?

$$0000\ 0110 = 4 + 2 = 6$$

$$1110\ 1111 = -(16 + 1) = -17$$

$$1111\ 1111 = -1$$

$$0101\ 0101 = 64 + 16 + 5 = 85$$

Problems 5-8

What are the decimal values of each of the following binary numbers if you interpret them as unsigned integers?

$$0000\ 0110 = 4 + 2 = 6$$

$$1110\ 1111 = (2^8 - 1) - 16 = 239$$

$$1111\ 1111 = 2^8 - 1 = 255$$

$$0101\ 0101 = 64 + 16 + 5 = 85$$

Problems 9-12

What are the decimal values of each of the following binary numbers if you interpret them as sign and magnitude integers?

$$0000\ 0110 = 6$$

$$1110\ 1111 = -(2^7 - 1 - 16) = -111$$

$$1111\ 1111 = -(2^7 - 1) = -127$$

$$0101\ 0101 = 64 + 15 + 5 = 85$$

Problems 12-16

What are the decimal values of each of the following binary numbers if you interpret them as 1's complement integers?

$$0000\ 0110 = 6$$

$$1110\ 1111 = -(0001\ 0000) = -16$$

$$1111\ 1111 = -0$$

$$0101\ 0101 = 85$$

Problems 17-20

Translate each of these to hexadecimal:

$$1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 0000\ 0110 = 0xFFFFFFFF06$$

$$1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1110\ 1111 = 0xFFFFFFFFEF$$

$$0111\ 1111\ 1111\ 1111\ 1111\ 0111\ 1110\ 1111 = 0x7FFFFFFEF$$

$$0101\ 0101\ 0101\ 0101\ 0101\ 0101\ 0101\ 0101 = 0x55555555$$

Problems 21-22

What is the largest positive number one can represent in a 15-bit 2's complement code? Write your result in binary and decimal.

$$011\ 1111\ 1111\ 1111$$

16383

Problems 23-24

What is the greatest magnitude negative number one can represent in a 15-bit 2's complement code? Write your result in binary and decimal.

100 0000 0000 0000
-16384

Problems 25-26

What is the largest positive number one can represent in a 15-bit 1's complement code? Write your result in binary and decimal.

011 1111 1111 1111
16383

Problems 27-28

What is the greatest magnitude negative number one can represent in a 13-bit 1's complement code? Write your result in binary and decimal.

1 0000 0000 0000
-4095

Problem 29

There are 124 students in CSE 30. If every student is to be assigned a unique bit pattern, what is the minimum number of bits required to do this?

$0111\ 1111 = 127$, so 7 bits is enough to represent each student.

Problem 30

How many more students can be accommodated in the class without requiring additional bits for each student's unique bit pattern?

7 bits can represent 0-127, so 128 students.
 $128 - 124 = 4$ students that can be added

Problem 31

How many bytes is required to store a 3 seconds of a 1080p video at 60 frames/sec? Assume that each pixel has three components Red, Green and Blue (RGB) and each of these colors can have a value between 0 and 255.

$(3 \text{ bytes/pixel (RGB)} * 2073600 \text{ pixels} * 3 \text{ seconds} * 60 \text{ frames/second}) = 1119744000 \text{ bytes}$

Problem 32

```

#include <stdio.h>
int main ( )
{
    unsigned int numToPrintInBase2 = 1431655765;
    /* alternating 1's and 0's */

    unsigned int exp = 1;
    int k; /* can't declare variable in a loop header */

    /* Compute the highest storable power
       of 2 (2 to the 31th). */
    for (k=0; k<31; k++)
    {
        exp = exp * 2;
    }

    /* For each power of 2 from the highest to the lowest,
       print 1 if it occurs in the number, 0 otherwise. */
// ERR- for (k=31; !(k=0); k--) not equals is improperly represented
    for(k=31; k!=0;k--)
    {
        if (numToPrintInBase2 >= exp)
        {
            //Err- printf ("%d", '1'); - would print ascii value
            printf("%d",1);
            numToPrintInBase2 = numToPrintInBase2 - exp;
        }
        else
        {
            // Err - printf ("%d", '0'); - would print ascii value
            printf("%d",0);
        }
        exp = exp / 2;
    }

    printf ("\n\n");
    return 0;
}

```

Problem 33

```
int bitCount (unsigned int n)
```

```
{
    int count1=0;
    while(n!=0)
    {
        if (n%2==1) count1++;
        n=n>>1;

    }

    return count1;
}
```

Problem 33

```
int parity(unsigned int n) {

    return (bitCount(n)%2);

}
```

Problem 34

```
int powerOfTwo(unsigned int) {

    //assuming 2^0 is a valid power
    if (bitCount(n)==1) return 1;
    //if 2^0 is not a valid power change it to :
    //if(bitCount(n>>1)==1) return 1;
    else return 0;

}
```