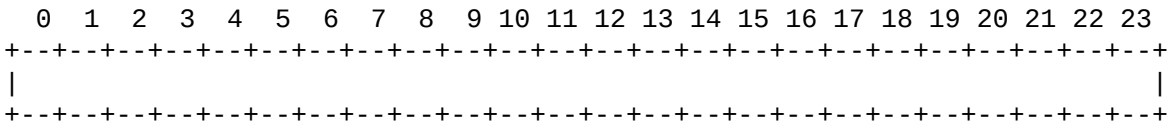
Example Floating Point Problems

Problem 1:

Consider the following program:

```
struct s {
    char c;
    double d;
    float f;
    short s;
};

union u {
    unsigned char buf[24];
    struct s a;
    int i;
} u1;

int main()
{
    int i,j;

    memset(&u1.a, 0, sizeof(struct s));

    u1.a.c = 0xab;
    u1.a.d = -3.5;
    u1.a.f = 0x1;
    u1.a.s = 0xcdef;
    u1.i = 0x12345678;

    /* print out the bytes of u1.buf as 2 digit
       hexidecimal numbers wit ha line break after
       every 8 bytes */
    for(i = 0; i < 3; i++)
    {
        for(j = 0; j < 8; j++)
           printf("0x\%.2x ",u1.buf[i*8+j]);
        printf("\n");
    }
}
```

You can use the following template as scratch space.



This program is compiled and run on a Linux/x86 machine. Fill in the output below. Write ?? if the value cannot be determined from the information provided.

0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____

0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____

0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____ 0x____

Problem 2:

Consider the following 7-bit floating point representation based on the IEEE floating point format:

- There is a sign bit in the most significant bit.
- The next $k = 3$ bits are the exponent. The exponent bias is 3.
- The last $n = 3$ bits are the fractional part.

Numeric values are encoded in this format as a value of the form $V = (-1)^s \times M \times 2^E$, where s is the sign bit, E is exponent after biasing, and M is the significand.

Part I

Answer the following problems using either decimal (e.g., 1.375) or fractional (e.g., $11/8$) representations for numbers that are not integers.

A. For denormalized numbers:

   (a) What is the value E of the exponent after biasing? _____

   (b) What is the largest value M of the significand? _____

B. For normalized numbers:

   (a) What is the smallest value E of the exponent after biasing? _____

   (b) What is the largest value E of the exponent after biasing? _____

   (c) What is the largest value M of the significand? _____

Part II

Fill in the blank entries in the following table giving the encodings for some interesting numbers.

Description	E	M	V	Binary Encoding
Zero		0	0	`0 000 0000`
Smallest Positive (nonzero)				
Largest denormalized				
Smallest positive normalized				
One			1	
Largest finite number				
NaN	—	—	NaN	
Infinity	—	—	$+\infty$	

Problem 3:

Consider a 8 bit floating point representation with a three-bit significand, four bits of exponent, a sign bit, and a bias value of 7. Assume that the implementation supports the IEEE standard (both normalized and denormalized values, and uses "nearest even" rounding). Fill in the empty boxes in the following table:

Description	Value	s	exponent	significand
zero	0.0			
closest positive to zero				
largest positive				
-5	-5.0			
	$2^{-4} - 2^{-6}$			
	$2.0 - \frac{9}{16}$			
	$\frac{6}{4} \cdot \frac{7}{4}$			

Recommended Book Practice Problems: 2.33, 2.34, 2.37
Solutions are at the end of the chapter.