CSE332/Floating Point Problems and Solutions

Prob-1: Convert -1313.3125 to IEEE 32-bit floating point format.

a. The integral part is $1313_{10} = 10100100001_2$. The fractional:

0.3125 × 2 = 0.625 0 Generate 0 and continue.
0.625 × 2 = 1.25 1 Generate 1 and continue with the rest.
0.25 × 2 = 0.5 0 Generate 0 and continue.
0.5 × 2 = 1.0 1 Generate 1 and nothing remains.

b. So $1313.3125_{10} = 10100100001.0101_2$.
c. Normalize: $10100100001.0101_2 = 1.01001000010101_2 \times 2^{10}$.
d. Mantissa is 01001000010101000000000,
e. exponent is 10 + 127 = 137 = 10001001_2,
f. sign bit is 1.

So -1313.3125 is 11000100101001000010101000000000

Sign bit	exponent	significant
1	10001001	01001000010101000000000

Prob-2: Convert 39887.5625 to IEEE 32-bit floating point format.

a. The integral part is $39887_{10} = 1001101111001111_2$. The fractional:

0.5625 × 2 = 1.125 1 Generate 1 and continue with the rest.
0.125 × 2 = 0.25 0 Generate 0 and continue.
0.25 × 2 = 0.5 0 Generate 0 and continue.
0.5 × 2 = 1.0 1 Generate 1 and nothing remains.

b. So $39887.5625_{10} = 1001101111001111.1001_2$.
c. Normalize: $1001101111001111.1001_2 = 1.0011011110011111001_2 \times 2^{15}$.
d. Mantissa is 00110111100111110010000,
e. exponent is 15 + 127 = 142 = 10001110_2,
f. sign bit is 0.

So 39887.5625 is 01000111000110111100111110010000

Sign bit	exponent	significant
0	10001110	00110111100111110010000

Prob-3: (a) The following numbers use the IEEE 32-bit floating-point format. What is the equivalent decimal value? i) 1 10000011 11000000000000000000000 ii) 0 01111110 10100000000000000000000 (b) Convert the following decimal number to IEEE 32-bit floating-point format i) -16.625 X 10 ^ 4 ii) -3013.3125

(a) i) 1 10000011 11000000000000000000000

$\text{Exponent} = (10000011)_2 = (131)_{10}$

$E' = E + 127$

$131 = E + 127$

$\boldsymbol{E} = \mathbf{4},$ the **base** is **2**

$(0.11)_2 = (1 * 2^{-1}) + (1 * 2^{-2}) = 0.75$; But '1.' Is **implicit** in IEEE-32bit. So, we add 1. So,

$(1 + 0.75) * 2^4 = 1.75 * 2^4 = (28)_{10}$

Since sign bit = 1, it is negative.

So, the equivalent decimal value is $= \mathbf{-28}$ **(Answer)**

ii) 0 01111110 10100000000000000000000

Exponent $= (01111110)_2 = (126)_{10}$

$E' = E + 127$

$126 = E + 127$

$\boldsymbol{E} = \mathbf{-1}$, the **base** is **2**

$(0.101)_2 = (1 * 2^{-1}) + (0 * 2^{-2}) + (1 * 2^{-3}) = (0.625)_{10}$. But ‘1.’ Is **implicit** in IEEE-32bit. So, we add 1. So,

$(1 + 0.625) * 2^{-1} = 0.8125 = 8.125\ X10^{-1}$

Since sign bit $= 0$, it is positive.

So, the equivalent decimal value is $= \mathbf{8.125X10^{-1}}$ **(Answer)**

(b) i) $\mathbf{-16.625X10^{-4} = -0.0016625}$

	0.0016625	X 2
0 ←	0.003325	X 2
0 ←	0.00665	X 2
0 ←	0.0133	X 2
0 ←	0.0266	X 2
0 ←	0.0532	X 2
0 ←	0.1064	X 2
0 ←	0.2128	X 2
0 ←	0.4256	X 2
0 ←	0.8512	X 2
1 ←	0.7024	X 2
1 ←	0.4048	X 2

0 ←	0.8096	X 2
1 ←	0.6192	X 2
1 ←	0.2384	X 2
0 ←	0.4768	X 2
0 ←	0.9536	X 2
1 ←	0.9072	X 2
1 ←	0.8144	X 2
1 ←	0.6288	X 2
1 ←	0.2576	X 2
0 ←	0.5152	X 2
1 ←	0.0304	X 2
0 ←	0.0608	X 2
0 ←	0.1216	X 2

	0.2432	X 2
0 ←	0.4864	X 2
0 ←	0.9728	X 2
1 ←	0.9456	X 2
1 ←	0.8912	X 2
1 ←	0.7824	X 2
1 ←	0.5648	X 2
1 ←	0.1296	X 2
0 ←	0.2592	

$(0.0016625)_{10} = (0.00000000011011001111010000111110)_2$
$= (1.1011001111010000011111 0\ X\ 2^{-10})_2$

$E' = E + 127$
$E' = -10 + 127 = 117$
$(117)_{10} = (0111\ 0101)_2$

Since the number is negative, sign bit = 1
So, the IEEE-32 floating point format is,

1 01110101 10110011110100000111110

ii) -3013.3125

2	3013	
2	1506 -	1
2	753 -	0
2	376 -	1
2	188 -	0
2	94 -	0
2	47 -	0
2	23 -	1
2	11 -	1
2	5 -	1
2	2 -	1
2	1 -	0
2	0 -	1

$(3013)_{10} = (101111000101)_2$
$(0.3125)_{10} = (0101)_2$

	0.3125
	X 2
0 ←	0.625
	X 2
1 ←	0.25
	X 2
0 ←	0.50
	X 2
1 ←	0.00

$(3013.3125)_{10} = (101111000101.0101)_2 = (1.011110001010101\ X\ 2^{11})_2$

$E' = E + 127$
$E' = 11 + 127 = 138$
$(138)_{10} = (1000\ 1010)_2$

Since the number is negative, sign bit = 1
So, the IEEE-32 floating point format is,

1 10001010 01111000101010100000000

Prob-4: Encode the decimal value +274.5625 as a 32-bit IEEE-754 floating point field and show your final answer in hexadecimal.

```
274.5625(10) = 100010010.1001(2)
Normalized:  1.000100101001 x 2**8

Mantissa part: .000100101001 (drop the leading 1.)       - pad on the
right with zeroes to fill up 23 bits:
         00010010100100000000000
Exponent part: 8
- excess-127 notation means add 127 before we convert to binary:
         8+127 = 135 = 128+7= 10000111(2)
     Sign: 0 (positive)

     In IEEE 754 single-precision (32-bit) format (1+8+23 bits):

      = 0 10000111 00010010100100000000000
      = 0100 0011 1000 1001 0100 1000 0000 0000
      =    4    3    8    9    4    8    0    0
      = 43894800h
```

Prob-5: Encode the decimal value -12.1875 as 32-bit IEEE-754 floating point field and show your answer in hexadecimal.

```
1. Number is negative so the sign bit will be 1

2. Convert 12.1875 to binary 1100.0011(2)

3. Normalize the binary number 1.1000011 * 2**3

4. The binary digits to the right of the decimal become the mantissa.
       Pad to the right with zeroes to fill up 23 bits:
       10000110000000000000000

5. The exponent is 3.  Bias it with 127 and it becomes 3+127 = 130.
       Convert 130 to binary becomes 10000010 (128+2)

6. Put it all together in 1+8+23=32 bits like this:
       = 1 10000010 10000110000000000000000
       = 1100 0001 0100 0011 0000 0000 0000 0000
       =    C    1    4    3    0    0    0    0
       = C1430000h
```

`Prob-6`: Convert the 32-bit floating point number to decimal.

01000100001101100001000000000000

a. Exponent: $10001000_2 = 136_{10}$; 136 − 127 = 9.
b. Denormalize: $1.01101100001_2 \times 2^9$ = 1011011000.01.
c. Convert:

Exponents	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}
Place Values	512	256	128	64	32	16	8	4	2	1	0.5	0.25
Bits	1	0	1	1	0	1	1	0	0	0 .	0	1
Value	512		+ 128	+ 64		+ 16	+ 8					+ 0.25 = 728.25

d. Sign: positive

Result: 728.25.

`Prob-7`: Convert the 32-bit floating point number to decimal.

10111110010110000000000000000000

a. Exponent: $01111100_2 = 124_{10}$; 124 − 127 = -3.
b. Denormalize: $1.1011_2 \times 2^{-3}$ = 0.0011011.
c. Convert:

Exponents	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
Place Values	1	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125
Bits	0 .	0	0	1	1	0	1	1
Value				0.125	+ 0.0625		+ 0.015625	+ 0.0078125 = 0.2109375

d. Sign: negative

Result: - 0.2109375

.

`Prob-8`: Convert the 32-bit floating point number a3358000 (in hex) to decimal.

10100011001101011000000000000000

a. Exponent: $01000110_2 = 70_{10}$; 70 − 127 = -57.
b. Since the exponent is far from zero, convert the original (normalized) mantissa:

Exponents	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	
Place Values	1	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125	0.00390625	
Bits	1 .	0	1	1	0	1	0	1	1	
Value	1		+ 0.25	+ 0.125		+ 0.03125		+ 0.0078125	+ 0.00390625	= 1.41796875

c. Sign: - (negative)

Result: - 1.41796875 × 2^{-57}

Prob-9: Convert the 32-bit floating point number to decimal.

$01110110011001010000000000000000_2$

a. Exponent: $11101100_2 = 236_{10}$; $236 - 127 = 109$.
b. Since the exponent is far from zero, convert the original (normalized) mantissa:

Exponents	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
Place Values	1	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.0078125
Bits	1 .	1	1	0	0	1	0	1
Value	1	+ 0.5	+ 0.25			+ 0.03125		+ 0.0078125 = 1.7890625

c. Number is 1.7890625×2^{109}.
d. Sign: positive

Result: 1.7890625×2^{109}

Prob-10: Convert -1313.3125 to IEEE 32-bit floating point format.

The integral part is $1313_{10} = 10100100001_2$. The fractional:

$0.3125 \times 2 = 0.625$ 0 Generate 0 and continue.
$0.625 \times 2 = 1.25$ 1 Generate 1 and continue with the rest.
$0.25 \times 2 = 0.5$ 0 Generate 0 and continue.
$0.5 \times 2 = 1.0$ 1 Generate 1 and nothing remains.

- So $1313.3125_{10} = 10100100001.0101_2$.
- Normalize: $10100100001.0101_2 = 1.01001000010101_2 \times 2^{10}$.
- Mantissa is 01001000010101000000000,
- exponent is $10 + 127 = 137 = 10001001_2$,
- sign bit is 1.

So -1313.3125 is 11000100101001000010101000000000

Sign bit	exponent	significant
1	10001001	01001000010101000000000

Prob-11: Convert 39887.5625 to IEEE 32-bit floating point format.

The integral part is $39887_{10} = 1001101111001111_2$. The fractional:

$0.5625 \times 2 = 1.125$ 1 Generate 1 and continue with the rest.
$0.125 \times 2 = 0.25$ 0 Generate 0 and continue.
$0.25 \times 2 = 0.5$ 0 Generate 0 and continue.
$0.5 \times 2 = 1.0$ 1 Generate 1 and nothing remains.

- So $39887.5625_{10} = 1001101111001111.1001_2$.
- Normalize: $1001101111001111.1001_2 = 1.0011011110011111001_2 \times 2^{15}$.
- Mantissa is 00110111100111110010000,
- exponent is $15 + 127 = 142 = 10001110_2$,
- sign bit is 0.

So 39887.5625 is 01000111000110111100111110010000

Sign bit	exponent	significant
0	10001110	00110111100111110010000

Prob-12: Encode the decimal value +274.5625 as a 32-bit IEEE-754 floating point field and show your final answer in hexadecimal.

```
274.5625(10) = 100010010.1001(2)
Normalized:  1.000100101001 x 2**8

Mantissa part: .000100101001 (drop the leading 1.)
- pad on the right with zeroes to fill up 23 bits:
         00010010100100000000000
Exponent part: 8
- excess-127 notation means add 127 before we convert to
binary:
         8+127 = 135 = 128+7= 10000111(2)
    Sign: 0 (positive)

    In IEEE 754 single-precision (32-bit) format (1+8+23
bits):

     = 0 10000111 00010010100100000000000
     = 0100 0011 1000 1001 0100 1000 0000 0000
     =    4    3    8    9    4    8    0    0
     = 43894800h
```

Prob-13: Encode the decimal value -12.1875 as 32-bit IEEE-754 floating point field and show your answer in hexadecimal.

```
1. Number is negative so the sign bit will be 1

2. Convert 12.1875 to binary 1100.0011(2)

3. Normalize the binary number 1.1000011 * 2**3

4. The binary digits to the right of the decimal become
the mantissa.
       Pad to the right with zeroes to fill up 23 bits:
       10000110000000000000000

5. The exponent is 3.  Bias it with 127 and it becomes
3+127 = 130.
       Convert 130 to binary becomes 10000010 (128+2)

6. Put it all together in 1+8+23=32 bits like this:
       = 1 10000010 10000110000000000000000
       = 1100 0001 0100 0011 0000 0000 0000 0000
       =    C    1    4    3    0    0    0    0
       = C1430000h
```