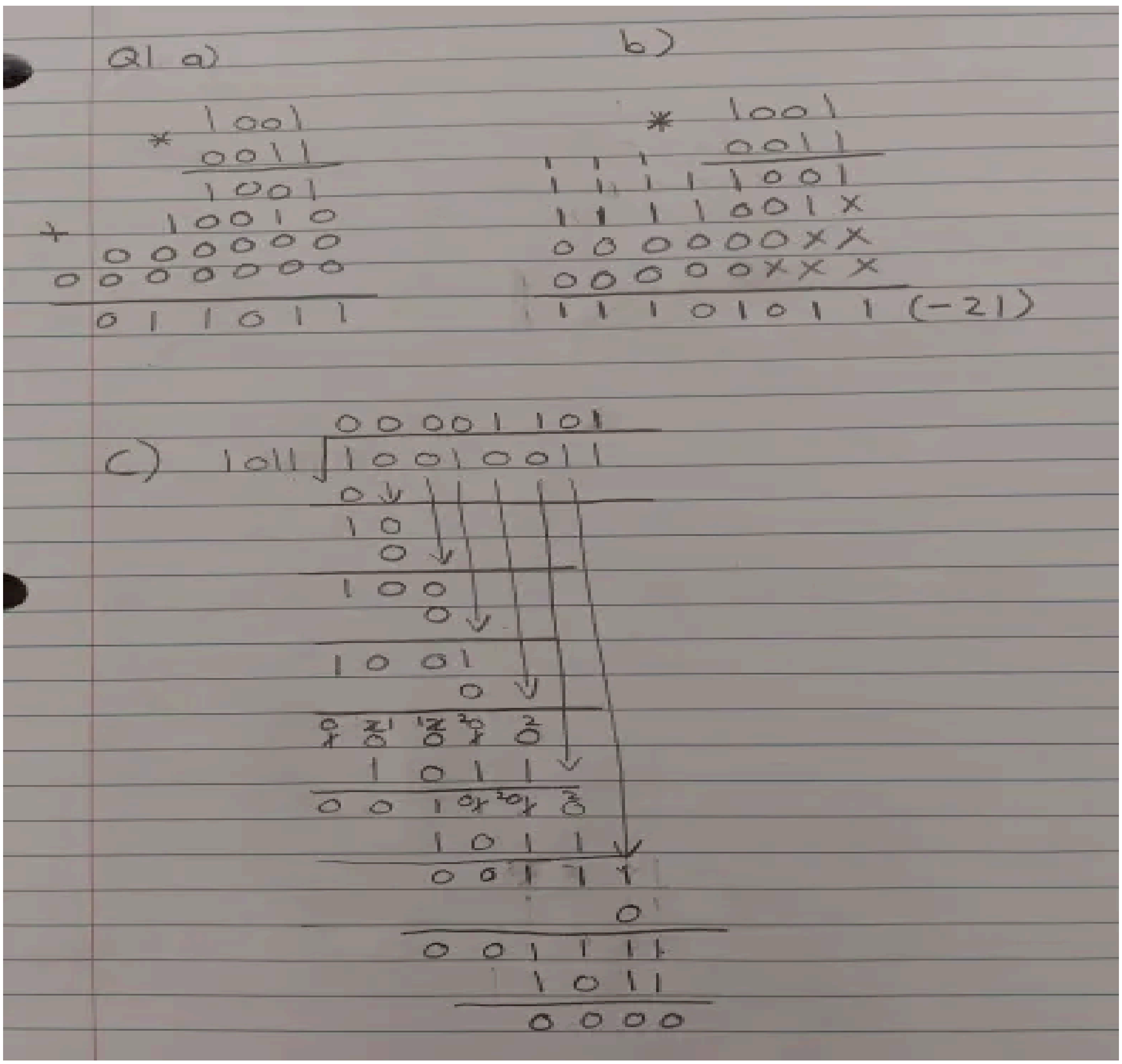
Q1. Perform the following arithmetic (showing all the steps used): (25 points)

a) Multiplication : 1001 X 0011 (unsigned integer) [Note. this is equal to 9 x 3)

b) Multiplications: 1001 x 0011 (two’s complement integers-signed number) [Note. this is equal to -7X3]

c) Division : 10010011 / 1011



A = 011011, B = 11101011, C = 01101

d) **Addition : 110111+1100**

e) **Subtraction: 39 – 25 (in binary). Remember. Subtraction in Binary is addition of the two's complement.**

Q1

d)

```
  111
  110111
+   1100
 -------
 1000011
```

e) 39 − 25

$39 \rightarrow 32+4+2+1$ $\quad \boxed{32}\ 16\ 8\ \boxed{4}\ \boxed{2}\ \boxed{1}$

1 0 0 1 1 1 = 39

39 − 32 = 7

7 − 4 = 3

4 − 2 = 2

2 − 1 = 1

$25 \rightarrow 16+8+1$ $\quad 32\ \boxed{16}\ \boxed{8}\ 4\ 2\ \boxed{1}$

1 1 0 0 1 = 25

25 − 16 = 9

9 − 8 = 1

Subtraction

```
  100111
 − 11001   = 1110
 -------
  001110
```

D = 1000011, E = 1110

Q2. Consider the following Floating-points in decimal: (40 pints)

$a = 125 \times 10^{-3}$, $b = 2.2 \times 10^{3}$, $c = 0.000625 \times 10^{2}$

A + B

Q2

$a = 125 \times 10^{-3}$, $b = 2.2 \times 10^{3}$, $C = 0.006625 \times 10^{2}$

Step 1 Align decimal point

$125 \times 10^{-3+3}$ $\quad$ $2.2 \times 10^{3-3}$

0.125×10^{0} $\quad$ 2200.0×10^{0}

Step 2. Add Significands

$0.125 + 2200.0 = 2200.125$

Step 3 Normalize

2200.125×10^{0}

$\therefore a + b = 2200 \times 10^{0}$

B - C

B − C

$b = 2.2 \times 10^3, \quad c = 0.000625 \times 10^2$

Align decimal:

$2.2 \times 10^{3-1} \qquad 0.000625 \times 10^2$

$22.0 \times 10^2 \qquad 0.000625 \times 10^2$

Subtract:

$22.0 - 0.000625$

$= 21.999375$

Round:

22.00

$\therefore \; B - C = 22.00 \times 10^2$

A * B

a × b

$a = 125 \times 10^{-3}, \quad b = 2.2 \times 10^3$

Align decimal:

$125 \times 10^{-3+6} \qquad 2.2 \times 10^3$

$0.000125 \times 10^3 \qquad 2.2 \times 10^3$

Multiply:

0.000125×2.2

$= 0.000275$

Round:

275×10^3

$\therefore \; a \times b = 275 \times 10^3$

A / C

a/c

$a = 125 \times 10^{-3}$, $c = 0.000625 \times 10^{2}$

Align Decimal:

$125 \times 10^{-3+5}$ $\quad$ 0.000625×10^{2}

0.00125×10^{2} $\quad$ 0.000625×10^{2}

Devide:

$0.00125 \div 0.000625$

$= 2$

Normalize:

2×10^{2}

$\therefore a/c = 2 \times 10^{2}$

a) Fill the following table by computing the required floating-point operations.

a + b	b - c	a x b	a/c
2200.125 X 10 ^0	22.00 X 10 ^2	275 X 10 ^3	2 X 10^2

Conversion to binary

A + B = 2200.125

b)

a+b = 2200.125

÷		
2	2200	Remainder
2	1100	0
2	550	0
2	275	0
2	137.5	1
2	68.5	1
2	34	0
2	17	0
2	8.5	1
2	4	0
2	2	0
2	1	0
2	0.5	1

0.125 × 2 = 0.25   0

0.25 × 2 = 0.5   0

0.5 × 2 = 1   1

2200 = 100010011000   0.125 = 001

∴

a+b = 100010011000.001

B - C = 22.00

b-c = 22.00

2	22	R
2	11	0
2	5.5	1
2	2.5	1
2	1	0
2	0.5	1

$\therefore$ 22 = 10110.00

A * B = 275

a x b

275

2	275	R
2	137.5	1
2	68.5	1
2	34	0
2	17	0
2	8.5	1
2	4	0
2	2	0
2	1	0
2	0.5	1

275 = 10010011

A / C = 2

$a/c = 2$

$$\begin{array}{c|cc|} 2 & 2 & \\ 2 & 1 & 0 \\ 2 & 0.5 & 1 \end{array}$$

$2 = 10$

b) :

a + b	b - c	a x b	a/c
100010011000.001	10110.00	10010011	10

c) For the following floating-point instruction: add.s $f1, $f2, $f3. Assume that the values of $f2, $f3 are a and b. What are the values (in binary) in $f2, $f3 and $f1 after executing this instruction?

c)

f2 = a $\quad$ $a = 125 \times 10^{-3}$

f3 = b $\quad$ $b = 2.2 \times 10^{3}$

Step 1 Align decimal point

$125 \times 10^{-3+3}$ $\quad$ $2.2 \times 10^{3-3}$

0.125×10^{0} $\quad$ 2200.0×10^{0}

Therefore $0.125 = f2$ $\quad$ $2200.0 = f3$

$0.125 + 2200.0 = 2200.125$

$\therefore$ f1 = 2200.125

Convertion to binary

f2	0.125	× 2 = 0.25	0	
	0.25	× 2 = 0.5	0	$\therefore$ f2 = 0.001
	0.5	× 2 = 1	1	

f3	2	2200.0		
	2	1100	0	
	2	550	0	
	2	275	0	
	2	137.5	1	$\therefore$ f3 = 100010011000
	2	68.5	1	
	2	34	0	
	2	17	0	
	2	8.5	1	
	2	4	0	
	2	2	0	
	2	1	0	
	2	0.5	1	

f1 = 2200.125

Based on f3 which is 2200
The binary representation is 100010011000

Based on f2 which is 0.125
the binary representation is 0.001

Therefore 2200.125 in binary is
100010011000.001

f1= 100010011000.001

$f1 =100010011000.001, $f2 = 100010011000, $f3 = 0.001

d) Consider a floating-point addition instruction add.d in MIPS assembly language, specifically focusing on the registers \$f2, \$f4, and \$f6. The task involves adding two double-precision floating-point values stored in registers \$f4 and \$f6, then storing the result in register \$f2.

1. Convert the decimal values of a and b into their IEEE 754 standard binary representation for double-precision floating-point format.

2. Execute the addition operation using the instruction add.d \$f2, \$f4, \$f6, where: o Register \$f4 contains the value of a. o Register \$f6 contains the value of b.

3. After performing the addition, provide the binary representation of the result stored in register \$f2 according to the IEEE 754 standard for double precision floating-point numbers.

d) $a = 125\times10^{-3}$, $b = 2.2\times10^{3}$

1. 125×10^{-3}

S = 0

Exponent = -3

fraction = 1111101 = 125

Bias: 1023

1023+3 = 1026

1026 = 10000000010

∴ Double Precision Format:

0 10000000010 1111101...000

S Exponent fraction

Convertion to Binary

		R
2	125	
2	62.5	1
2	31	0
2	15.5	1
2	7.5	1
2	3.5	1
2	1.5	1
2	0.5	1

125 = 1111101

		R
2	1026	
2	513	0
2	256.5	1
2	128	0
2	64	0
2	32	0
2	16	0
2	8	0
2	4	0
2	2	0
2	1	0
2	0.5	1

1026 = 10000000010

b = 2.2×10^3

2.2×10^3
S = 0
Exponent = 3
Fraction = 2.2 = 00100
Bias = 1023
1023 + 3 = 1026
1026 = 10000000010

∴ Double Precision format

0 10000000010 00100...110
S Exponent fraction

Convertion to Binary:

2	2	R
2	1	0
2	0.5	1

2 = 10

0.2 × 2 = 0.4	0	0.2 = 00100
0.4 × 2 = 0.8	0	
0.8 × 2 = 1.6	1	
0.6 × 2 = 1.2	1	
0.2 × 2 = 0.4	0	
0.4 × 2 = 0.8	0	

2.
(f4) a = 0 10000000010 1111101000...
(f6) b = 0 10000000010 0011001100...

 0100000000010111111010000
+ 0100000000010 0011001100

= 2.000000000020...

∴ f2 = 2.000000000020

3. f2 = 2.00000000020

S = 0
Exponent = 1
fraction = 2.0000000020
= 10.0011001100...
Bias = 1023
1023 + 1 = 1024
1024 = 10000000000

∴ Double Precision format

0 10000000000 10001100110...
S Exponent fraction

Convertion
2 = 10

0.20 × 2 = 0.4	0
0.40 × 2 = 0.8	0
0.8 × 2 = 1.6	1
0.6 × 2 = 1.2	1
0.2 × 2 = 0.4	0

0.000000000020
= 00110...

∴ f3 = 0 10000000000 1000110...

2	1024	R
2	512	0
2	256	0
2	128	0
2	64	0
2	32	0
2	16	0
2	8	0
2	4	0
2	2	0
2	1	0
2	0.5	1

1024 = 10000000000

F4 = 0100000000101111101000…, F6 = 0100000000100011001100…,
F2= 01000000000010001100…

Q3. In Floating Point representations, answer the following questions showing all the steps: (Use the IEEE 754 standard for all) (showing all the steps used): (35 points)

a) Write down the binary representation of the decimal number 31.625

a) 31.625

Step1

÷		R
2	31	
2	15.5	1
2	7.5	1
2	3.5	1
2	1.5	1
2	0.5	1

$0.625 \times 2 = 1.25$ 1
$0.25 \times 2 = 0.5$ 0
$0.5 \times 2 = 1$ 1

$\therefore 31 = 11111$ $\therefore 0.625 = 101$

Step 2. $S = 0$

fraction: $31.625 = 11111.101$
(shift 4 places left)

$\therefore 1.1111101\ldots \times 2^4$

Step 3
Exponent:
Based on the fraction, the exponent is 4

$\therefore 127+4 = 131$

Step 4

÷		R
2	131	
2	65.5	1
2	32.5	1
2	16	0
2	8	0
2	4	0
2	2	0
2	1	0
2	0.5	1

$131 = 10000011$

$\therefore$ Single precision format = 0 10000011 1111101…
(S) (exponent) (fraction)

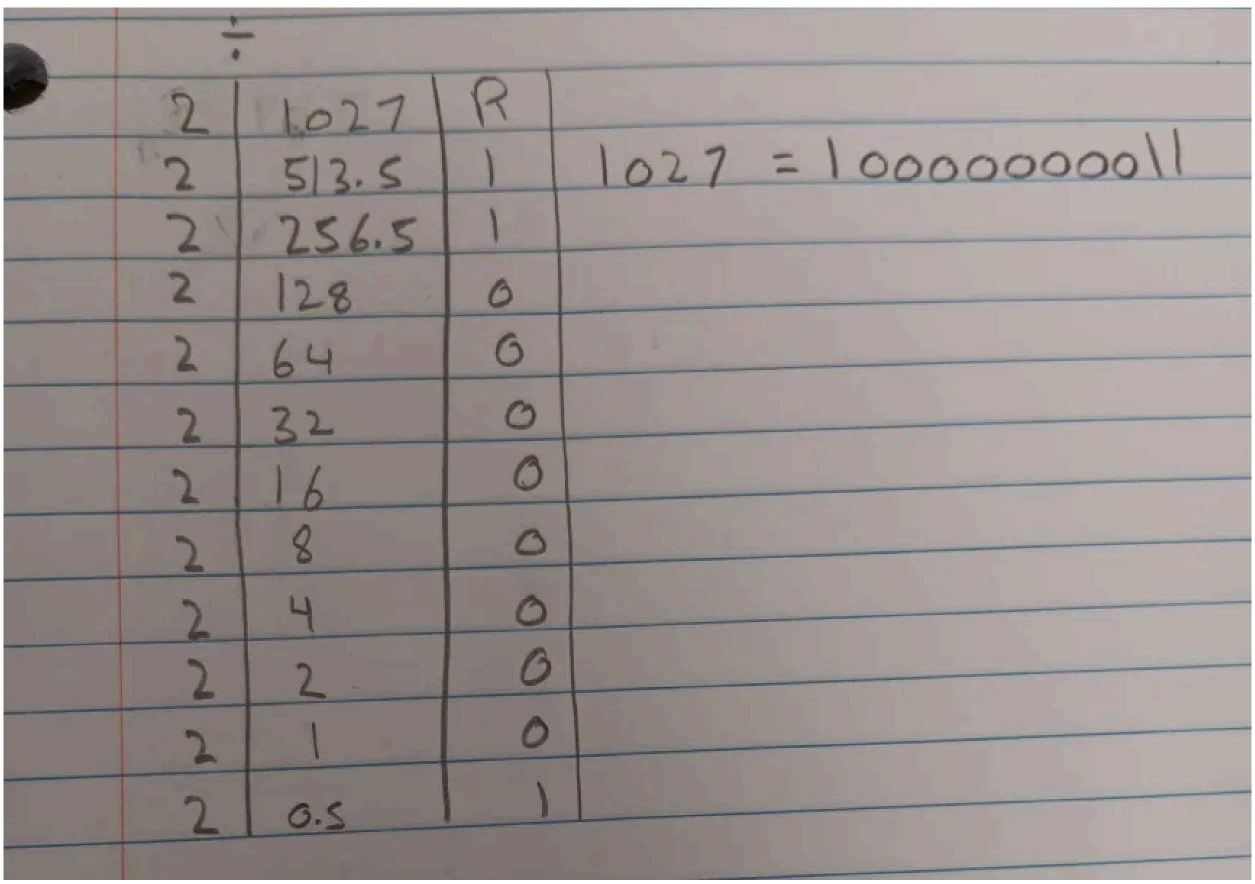
Single precision format full number: 0 10000011 11111101000000000000000

32 bit

b) Write down the binary representation of the decimal number 31.625 assuming the IEEE 754 double precision format.

Based on the work shown in question a, the fraction value is 11111101, the exponent is 4. The bias value for double precision format is 1023. Therefore we would add the bias and exponent (4) shown in question a. So, 1023 + 4 = 1027.

1027 converted to binary



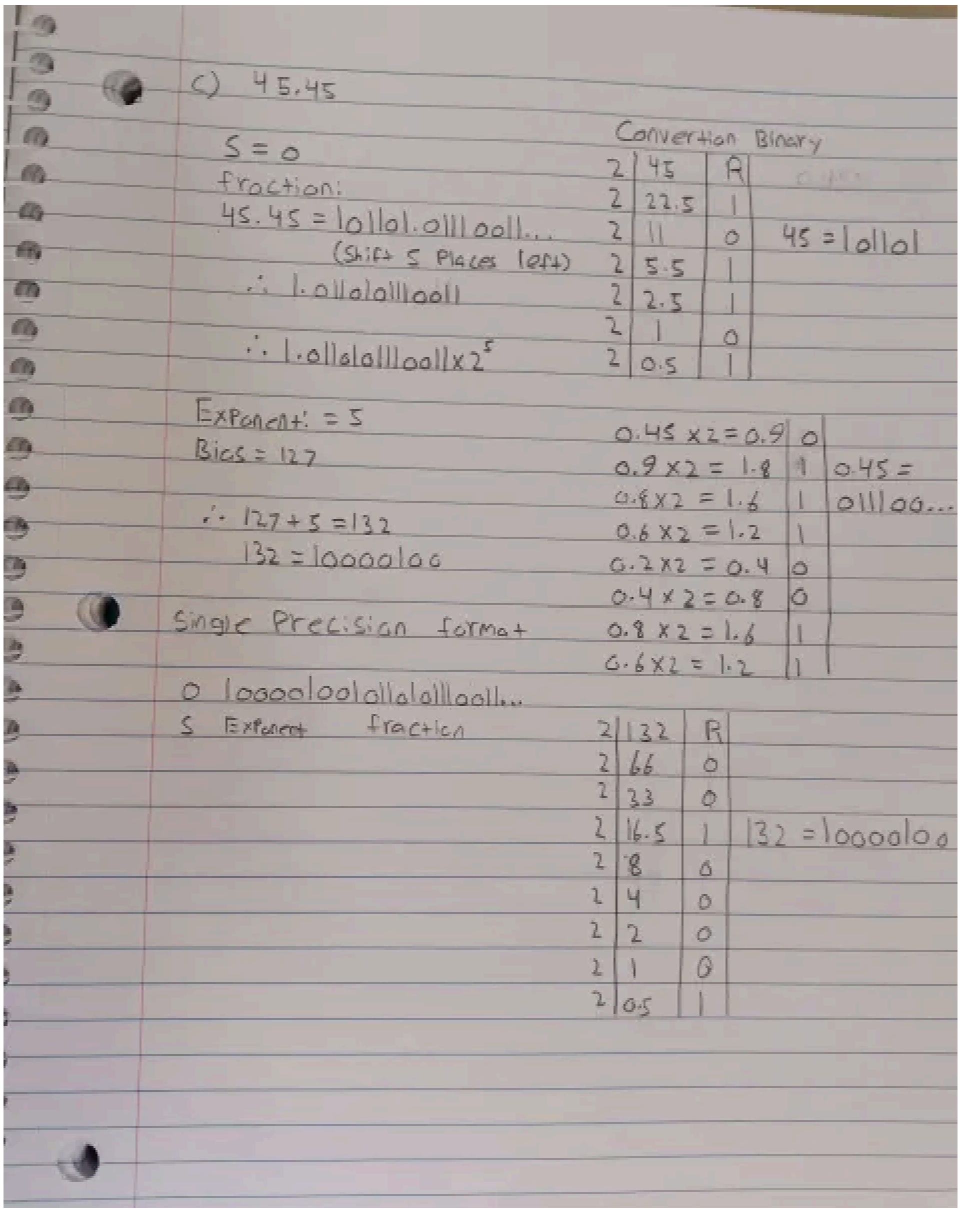
So now we have everything we need to put this into the double precision format. So

S for this its 0 because its positive:

Answer in double ieee 754 precision format:

0 10000000011 11111101…00 (64 bit)

c) Write down the binary representation of the decimal number 45.45.



c) 45.45

S = 0

fraction:

45.45 = 101101.01110011...

(Shift 5 places left)

∴ 1.0110101110011

∴ $1.0110101110011 \times 2^5$

Exponent: = 5

Bias = 127

∴ 127 + 5 = 132

132 = 10000100

Single Precision format

0 1000010001101011100 11...

S Exponent fraction

Conversion Binary

		R
2	45	
2	22.5	1
2	11	0
2	5.5	1
2	2.5	1
2	1	0
2	0.5	1

45 = 101101

0.45 x 2 = 0.9	0
0.9 x 2 = 1.8	1
0.8 x 2 = 1.6	1
0.6 x 2 = 1.2	1
0.2 x 2 = 0.4	0
0.4 x 2 = 0.8	0
0.8 x 2 = 1.6	1
0.6 x 2 = 1.2	1

0.45 = 011100...

		R
2	132	
2	66	0
2	33	0
2	16.5	1
2	8	0
2	4	0
2	2	0
2	1	0
2	0.5	1

132 = 10000100

Full answer: 0 1000100 10110101110011001101011 (32 bit)

Double precision format:

(bias) 1023 + 5 = 1028

Convert the binary

1028 / 2 = 514 R 0

512 / 2 = 256 R 0

256 / 2 = 128 R 0

128 / 2 = 64 R 0

64 / 2 = 32 R 0

32 / 2 = 16 R 0

16 / 2 = 8 R 0

8 / 2 = 4 R 0

4 / 2 = 2 R 0

2 / 2 = 1 R 0

1 / 2 = 0.5 R 1

1028 = 1000000000

Double precision =

0 1000000000 10110101110011…0011 (64 bit)

d) Write down the binary representation of the decimal number -19.625.

d) −19.625

S = 1

fraction:

−19.625 = −10011.101

(Shift 4 places left)

$\therefore = -1.0011101 \times 2^4$

Exponent: = 4

Bias = 127

$\therefore$ 4 + 127 = 131

$\therefore$ Single precision format

1 1000001110011101...

(S) (Exponent) (fraction)

Conversion Binary

2	19	R
2	9.5	1
2	4.5	1
2	2	0
2	1	0
2	0.5	1

19 = 10011

0.625 × 2 = 1.25	1
0.25 × 2 = 0.5	0
0.5 × 2 = 1	1

0.625 = .101

2	131	R
2	65.5	1
2	32.5	1
2	16	0
2	8	0
2	4	0
2	2	0
2	1	0
2	0.5	1

131 = 10000011

Full answer: 1 1000011 10011101000000000000000 (32 bit)

Double precision format:

(Bias) 1023 + 4 = 1027

Conversion to binary

1027 / 2 = 513.5 R 1

513 / 2 = 256.5 R 1

256 / 2 = 128 R 0

128 / 2 = 64 R 0

64 / 2 = 32 R 0

32 / 2 = 16 R 0

16 / 2 = 8 R 0

8 / 2 = 4 R 0

4 / 2 = 2 R 0

2 / 2 = 1 R 0

1 / 2 = 0.5 R 1

1027 = 10000000011

Double precision format:

1 10000000011 10011101…000 (64 bit)

e) What decimal number does the bit pattern 0×0C000000 represent if it is a floating-point number?

According to the hexadecimal binary conversion table in the textbook:

Hexadecimal	Binary	Hexadecimal	Binary	Hexadecimal	Binary	Hexadecimal	Binary
0_{hex}	0000_{two}	4_{hex}	0100_{two}	8_{hex}	1000_{two}	c_{hex}	1100_{two}
1_{hex}	0001_{two}	5_{hex}	0101_{two}	9_{hex}	1001_{two}	d_{hex}	1101_{two}
2_{hex}	0010_{two}	6_{hex}	0110_{two}	a_{hex}	1010_{two}	e_{hex}	1110_{two}
3_{hex}	0011_{two}	7_{hex}	0111_{two}	b_{hex}	1011_{two}	f_{hex}	1111_{two}

FIGURE 2.4 The hexadecimal-binary conversion table.

The bit pattern 0×0C000000 represents

0 00001100 00000000000000000000000

The first 0 is the sign bit indicating its a positive number

The exponent is: 000011000

The significand value is: 00000000000000000000000

0 00001100 00000000000000000000000

S=0

Fraction: 000000...

Exponent: 00001100

Convert to decimal

2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0

0 0 0 0 1 1 0 0

$2^3 + 2^2 = 12$

$\therefore$ 00001100 = 12

Formula:

$(-1)^S \times (1 + \text{Fraction}) \times 2^{(\text{Exp}-\text{bias})}$

$(-1)^0 \times (1 + 0) \times 2^{(12-127)}$

$1 \times 1 \times 2^{-115}$

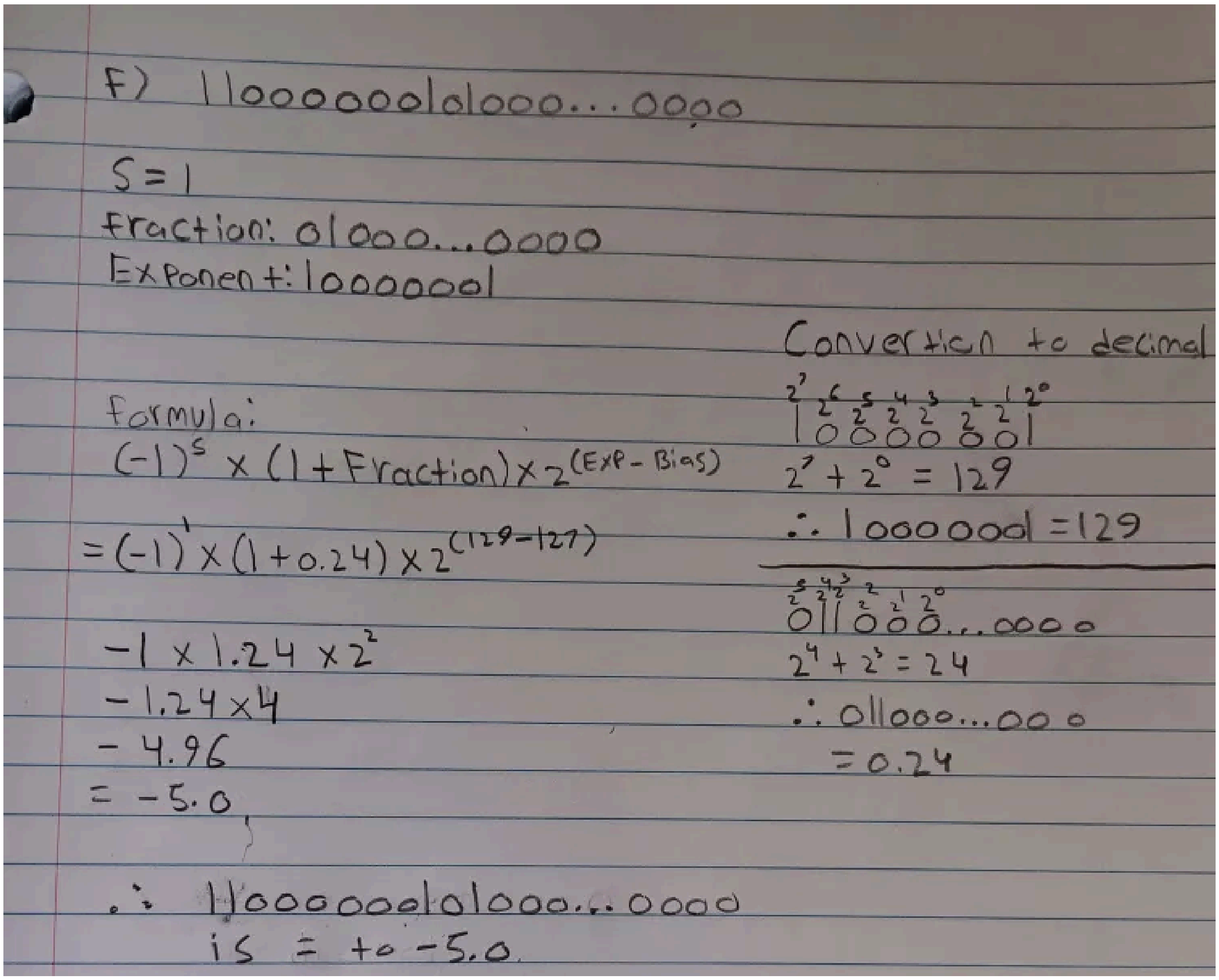
1 x 2.407

=2.407

Round: 2

= 2.4

f) What number is represented by the following single precision float (single Precision) : 11000000101000…0000



F) 11000000101000...0000

$S = 1$

Fraction: 01000...0000

Exponent: 10000001

Convertion to decimal

$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

10000001

$2^7 + 2^0 = 129$

∴ 10000001 = 129

$2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

011000...0000

$2^4 + 2^3 = 24$

∴ 011000...000

$= 0.24$

formula:

$(-1)^S \times (1 + \text{Fraction}) \times 2^{(\text{Exp} - \text{Bias})}$

$= (-1)^1 \times (1 + 0.24) \times 2^{(129-127)}$

$-1 \times 1.24 \times 2^2$

-1.24×4

-4.96

$= -5.0$

∴ 11000000101000...0000 is = to -5.0

= - 5.0

g) What number is represented by the following single precision float (single Precision) :11000000111000000000000000000…0000

g) 11000000111000000000000000000...0000

S = 1

fraction = 10000000000000000000...0000

Exponent = 10000011

formula:

$(-1)^S \times (1 + \text{fraction}) \times 2^{(EXP-127)}$

$(-1)^1 \times (1 + 1000000000000000000.0000) \times 2^{(131-127)}$

$-1 \times (1.0000\ldots) \times 2^4$

-1×16

-16

∴ The answer is -16

Conversion to decimal:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	0	0	0	0	0	1	1

$2^0 + 2^1 + 2^7 = 131$

$10000011 = 131$

= -16