Capítulo 3: TEORIA DE PUNTO FLOTANTE

3.1 Introducción

Se denominan ***Números en Punto Flotante*** a las representaciones internas del procesador que modelan a los números reales. En forma externa, se representan números con punto decimal tal como 3.1415926 o en notación científica 9.512×10^{-5}, con un solo dígito a la izquierda del punto decimal; es decir, 9.512×10^{-4}.

3.2 Normalización de números

Se dice que el número está normalizado si el dígito a la izquierda del punto no es cero. En el ejemplo anterior, 95.12×10^{-5}. De manera normalizada quedaría: 9.512×10^{-4}. En el caso de computadores, se emplea números en sistema binario, y con un número finito de dígitos.

3.3 Propósito de números en punto flotante

Como la memoria de los computadores es limitada, no puedes almacenar números con precisión infinita, no importa si usas fracciones binarias o decimales, en algún momento tienes que cortar. Pero ¿cuánta precisión se necesita? ¿Y *dónde* se necesita? ¿Cuántos dígitos enteros y cuántos fraccionarios?

La idea es descomponer los números en dos partes:

- Una **mantisa** (también llamada coeficiente o significando) que contiene los dígitos del número, donde las mantisas negativas representan números negativos; es decir, la mantisa corresponde a la parte fraccionaria que es la diferencia entre el número y la parte entera del número. Veamos este ejemplo, en el número decimal 13.8543, la parte entera es 13 y la mantisa=13.8543-13= 0,8543. Pero, cuando el número decimal es negativo; esto es, -13.8543, la parte entera es -14 y la mantisa=-13.8543-(-14)= 0.1457.
- Un **exponente** que indica dónde se coloca el punto decimal (o binario) en relación al inicio de la mantisa. Cuando el exponente es negativo representará a un número menor que uno.

Este formato cumple todos los requisitos:

- Puede representar números de órdenes de magnitud enormemente dispares (limitado por la longitud del exponente).
- Proporciona la misma precisión relativa para todos los órdenes (limitado por la longitud de la mantisa).
- Permite cálculos entre magnitudes: multiplicar un número muy grande y uno muy pequeño conserva la precisión de ambos en el resultado.
- Los números de punto flotante decimales normalmente se expresan en notación científica con un punto explícito siempre entre el primer y el segundo dígitos. El

exponente o bien se escribe explícitamente incluyendo la base, o se usa una **E** para separarlo de la mantisa.

Ejemplo 3.1: según la parte entera, la mantisa y el exponente de la tabla de la figura 3.1, escriba su valor equivalente, en notación científica y su valor en punto flotante.

Parte entera	Mantisa	Exponente	Notación científica	Valor en punto fijo
1	0.5	4	1.5×10^4	15000
5	0	-3	5×10^{-3}	0.005
6	0.667	-11	6.667E-11	0.0000000000667
-3	0.999	3	-3.999×10^3	-3999

Tabla 1: formato de configuración de punto flotante

3.4 Cifras significativas

Las *cifras significativas* representan el uso de una o más escalas de incertidumbre en determinadas aproximaciones. Se dice que 4.7 tiene 2 cifras significativas, mientras que 4.70 tiene 3. Para distinguir los ceros que son significativos de los que no lo son, estos últimos suelen indicarse como potencias de 10 en notación científica, por ejemplo 5724 será 5.724×10^3, con 4 cifras significativas.

Cuando no pueden ponerse más cifras significativas, cierta cantidad de ellas, por ejemplo de tres cifras simplemente, a la tercera cifra se le incrementa un número, si la cifra predecesora es 5 con otras cifras o mayor que 5. Si dicha cifra es menor simplemente se deja igual. Ejemplo 5.3689 consta de 5 cifras significativas, si sólo se pueden mostrar tres cifras, se le suma una unidad a la cifra 6 (6+1=7) ya que la cifra que la precede 8 es mayor que 5, así que queda 5.37 y si el número es menor que cinco: así 5.36489 y se redondea queda 5.36, no aumenta por que la cifra 4 es menor que 5.

Cuando la cifra a redondear está precediendo a 5, simplemente aumente en 1 dicha cifra. Por ejemplo para redondear a 3 cifras los números 12.45 y 0.1855, se observa que en el primer número, el dígito 3 que precede al 5, se incrementa en 1 cifra quedando 12.5 y el segundo número 0.1855, se mantiene su valor, 0.186 o 1.86×10^{-1}.

El uso de las cifras significativas considera que el último dígito de aproximación es incierto, por ejemplo, al determinar el volumen de un líquido con una probeta cuya resolución es de 1 ml, implica una escala de incertidumbre de 0.5 ml. Así se puede decir que el volumen de 6 ml será realmente de 5.5 ml a 6.5 ml. El volumen anterior se representará entonces como (6.0 ± 0.5) ml. En caso de determinar valores más próximos se tendrían que utilizar otros instrumentos de mayor resolución, por ejemplo, una probeta de divisiones más finas y así obtener (6.0 ± 0.1) ml o algo más satisfactorio según la resolución requerida.

3.5 Estándar IEEE 754

El *estándar IEEE 754* ha sido definido por el Instituto de Ingenieros Eléctricos y Electrónicos (*Institute of Electrical and Electronics Engineers*, *IEEE*) y establece dos formatos básicos para representar a los números reales en la computadora digital: precisión simple y precisión doble.

Formato	Bits totales	Bits significativos	Bits del exponente	Número más pequeño	Número más grande
Precisión simple	32	23 + 1 signo	8	~1.2×10^{-38}	~3.4×10^{38}
Precisión doble	64	52 + 1 signo	11	~5.0×10^{-324}	~1.8×10^{308}

Tabla 2: estándar de IEEE 754

La mantisa, normalmente se utiliza Signo Magnitud. Además, la mantisa se suele normalizar colocando el punto decimal a la derecha del bit más significativo.

3.6 Precisión Simple en el Estándar IEEE 754

Para escribir un número en el Estándar IEEE 754 en precisión simple, se escribe el número real usando 32 bits (4 bytes): 1 bit para el signo (s) del número, 23 bits para la mantisa (m) y 8 bits para el exponente (E), que se distribuyen de la siguiente forma:

31	30 . . . 23	22 . . . 0
S	E	m
signo	Exponente	Mantisa

E= exponente representación externa + exponente en exceso.

La representación del exponente a exceso es $2^{n-1}-1$, mientras que, el exponente externo resulta, cuando se normaliza el número (el exponente de 10 o de 2).
En la siguiente tabla se resumen los cálculos que hay que realizar para deducir el valor en base 10 de un número escrito en el estándar IEEE 754 con precisión simple:

Signo(S)	Exponente(E)	Mantisa(m)	Valor decimal
0 ó 1	1<E<255	Cualquiera	$(-1)^s * 1.m * 2^{E-127}$
0	E=255	m=0	$+\infty$
1	E=255	m=0	$-\infty$
0 ó 1	E=255	$m \neq 0$	NaN
0 ó 1	E=0	m=0	0
0 ó 1	E=0	$m \neq 0$	$(-1)^s * 0.m * 2^{-126}$

Figura 3: representación de un número real con precisión simple en el estándar IEEE 754.

Ejemplo 3.2: exprese en formato IEEE 754 de precisión simple, el valor del número 45.25
Solución
Pasando la mantisa a base 2 y normalizando, se tiene: $45=101101_2$ y $0.25=0.01_2$
$45.25=45+0.25=101101_2 + 0.01_2=101101.01_2=1.0110101_2 * 2^5$

Veamos cómo se obtiene el valor del número:

- Como el signo es positivo, el valor del signo es 0.
- Representando en forma binaria el exponente en exceso 2^{n-1}-1 donde n (cantidad de cifras significativas) es 8 y e (exponente representación externa) que es 5 en este ejemplo. Por lo tanto, el exponente en exceso es 2^{8}-1-1=127.

Por consiguiente:

$E=5+127=132=10000100_2$
101101.01_2
El bit omitido que antecede a la mantisa (luego de normalizado el número binario) se le llama bit implícito. Por otra parte, el bit de signo vale 0, ya que, el número es positivo. Por lo tanto, eliminando el bit implícito de la mantisa (1**.**0110101_2x2^5) esta quedaría: m=0110101=01101010000000000000000

- Por lo tanto, la representación final corresponde

Signo	Exponente (E)	Mantisa (m)
0	10000100	01101010000000000000000

Ejemplo 3.3: Para escribir el número
101110**.**$010101110100001111100001111100010011_2$
en el estándar IEEE 754 con precisión simple, con exponente en exceso a 2^{n-1}-1 y mantisa m y signo s. Determine además, su número hexadecimal correspondiente
Primero tendremos que normalizarlo:
101110**.**$010101110100001111100001111100010011_2$
=1**.**$01110010101110100001111100001111100010011_2$ x 2^5
El exponente E será: exponente representación externa más el exponente en exceso a 2^{n-1}-1 en base 10 en precisión simple es 127. Calculemos ahora a E:
$E=5 + (2^{8-1}-1)=5 + (2^7-1)=5 + (128-1)=132=10000100$
De la mantisa solo se cogerán los 23 bits más significativos:
1**.**0111001010111010000111
Como el resto de bits no pueden representarse, ya que no caben en la mantisa, entonces se descartarán. Sin embargo, cuando la mantisa se normaliza situando el punto decimal a la derecha del bit más significativo, dicho bit siempre vale 1. Por lo tanto, se puede prescindir de él, y coger en su lugar un bit más de la mantisa de la parte descartada. De esta forma, la precisión del número representado es mayor. Así, los bits de la mantisa serán:
01110010101110100001111
En consecuencia, el número se puede representar como:

31	30 … 23	22 . . . 0
0	10000100	01110010101110100010000
Signo	Exponente	Mantisa

Los programadores, para representar los números reales en este formato, suelen hacerlo en el Sistema Hexadecimal; efectivamente, tome toda la cadena de bits y agrúpelos de a 4 bits para representar cada cifra hexadecimal.

Así pues, **101110**.$010101110100001111100001111100010011_2$
$101110.010101110100001111100001111100010011_2$
=1.0111001010111010000100001~~00001111100010011~~$_2$x2^5

Ahora, determinemos la mantisa a partir de $1.01110010101110100010000_2 x 2^5$ los demás bits se descartan.

$\Rightarrow$ mantisa m= 01110010101110100001111; signo s=0; exponente E=132=10000100;

4 2 3 9 5 D 0 F

0100 0010 0011 1001 0101 1101 0001 0000

Por lo tanto, el número en precisión simple corresponderá a:
0 10000100 $01110010101110100010000_2 x 2^5$=$42395D10_H$.

En este caso, los números no son exactamente iguales, ya que con precisión simple no han podido representarse todos los bits de la mantisa.

Ejemplo 3.4: Dado el número $3E400000_H$ del estándar IEEE 754 con precisión simple, exponente en Exceso a 2^{n-1}-1 y mantisa m con 1 bit implícito, signo s, averigüe a qué número representa en base 10.
En efecto, pueden realizarse los siguientes pasos:

1º) Convertir $3E400000_H$ a base 2:

3 E 4 0 0 0 0 0

0011 1110 0100 0000 0000 0000 0000 0000

0 01111100 10000000000000000000000

2º) Obtener los bits del signo, de la mantisa y del exponente:
Signo (positivo)=0; n (8 bits)=01111100; mantisa (23 bits)= 10000000000000000000000

31	30 . . . 23	22 . . . 0
0	01111100	10000000000000000000000
Signo	Exponente	Mantisa

3º) Pasar el exponente a base 10:
$(01111100_2 - (2^{8-1} - 1)) = 124 - (2^7 - 1)$
$= 124 - (128 - 1) = 124 – 127 = -3$

4º) Escribir el número en notación científica. Para ello, la mantisa se debe escribir con el bit implícito (1), seguido del punto decimal (.) y de los bits de la mantisa (10000000000000000000000), teniendo en cuenta que los ceros por la derecha se pueden despreciar. Por otra parte, el número es positivo, ya que el bit de signo es 0. Por tanto, el número es: $1.10000000000000000000000 x 2^{-3}$=$1.1 \times 2^{-3}$

5º) Expresar el número en base 10.
$1.1x\ 2^{-3} = ((2^0 + 2^{-1}) \times 2^{-3}) = ((1 + 0.5) \times 0.125) = (1.5 \times 0.125) = 0.187510$
Por tanto, $3E400000_H$ (precisión simple) = 1.1×2^{-3} = +**0.187510**

3.8 Precisión Doble en el Estándar IEEE 754

Para escribir un número real en precisión doble, se emplean 64 bits (8 bytes): 1 bit para el signo (s) del número, 52 bits para la mantisa (m) y 11 bits para el exponente (E).

63	62 . . . 52	51 . . . 0
S	E	M
signo	Exponente	Mantisa

En la siguiente tabla se resumen los cálculos que hay que realizar para deducir el valor en base 10 de un número escrito en el estándar IEEE 754 con precisión doble:

Signo(S)	Exponente(E)	Mantisa(m)	Valor decimal
0 ó 1	1<E<2047	Cualquiera	$(-1)^{s} * 1.m * 2^{E-1023}$
0	E=2047	m=0	$+\infty$
1	E=2047	m=0	$-\infty$
0 ó 1	E=2047	m≠0	NaN
0 ó 1	E=0	m=0	0
0 ó 1	E=0	m≠0	$(-1)^{s} * 0.m * 2^{-1022}$

Figura 3.4: representación de un número real con precisión doble en el estándar IEEE 754.

Ejemplo 3.5: Si se quiere escribir el número 19.5625 en el estándar IEEE 754 con precisión doble, exponente en Exceso a 2^{n-1}-1 y mantisa m con 1 bit implícito, signo s, dado en hexadecimal.

Los pasos a seguir son:

1º) Cambiar 19.5625 a base 2. Primero la parte entera:
19 DIV 2=9 y 19 Mod 2=1
9 DIV 2= 4 y 9 Mod 2= 1
4 DIV 2= 2 y 4 Mod 2= 0
2 DIV 2=1 y 2 Mod 2= 0
1 DIV 2 =0 y 1 Mod 2= 1

⇨ $19=10011_2$

y, a continuación, la parte fraccionaria:
0.5625x2=1.125 =>1
0.125x2=0.25 => 0
0.25x2=0.5 => 0
0.5x2=1.0 => 1

⇨ $0.5625=1001_2$

De modo que, $19.5625 = 10011.1001_2$

2º) Normalizar el número binario obtenido, colocando el punto decimal a la derecha del bit más significativo: $10011.1001_2 = 1.00111001 \times 2^4$

3º) Escribir en base 10 el exponente en Exceso a $2^{n-1} - 1$, donde n=11: $4 + (2^{11-1} - 1 = 4 + (2^{10} - 1) = 4 + (1024 - 1) = 1027 = 10000000011_2$ Exponente en exceso a 1023

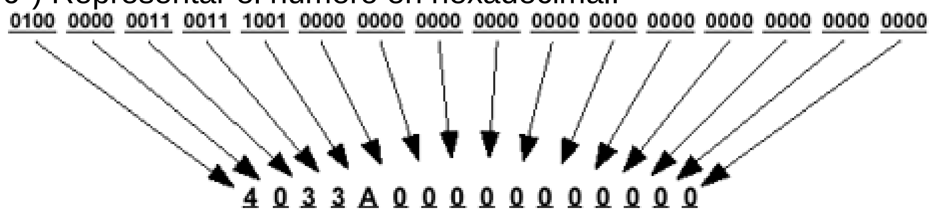
4º) Establecer la mantisa utilizando bit implícito. Para ello, se cogen los ocho bits que están a la derecha del punto (00111001) y el resto de la mantisa (52 bits) se rellena con ceros:

0011100100000000000000000000000000000000000000000000

5º) Expresar el número en el estándar IEEE 754 con precisión doble. En este caso, hay que tener en cuenta que el bit de signo vale 0, ya que, el número es positivo:

63	62 . . . 52	51 . . . 0
0	10000000011	0011100100000000000000000000000000000000000000000000
signo	Exponente	Mantisa

6º) Representar el número en hexadecimal:



De tal forma que, 19.5625 = 10011.1001_2 = 1.00111001 x 2^4 = **$4033A000000000000_H$** (precisión doble)

Casos especiales en el Estándar IEEE 754

Tanto en precisión doble como en precisión simple, existen algunos casos especiales que dependen de los valores del signo, del exponente y de la mantisa:

Signo (s)	*Exponente (exp)*	*Mantisa (m)*	*Significado*
Positivo (0)	Todo unos (11...1)	Todo ceros (00...0)	Más infinito (+∞)
Negativo (1)	Todo unos (11...1)	Todo ceros (00...0)	Menos infinito (-∞)
0 ó 1	Todo unos (11...1)	Distinta de todo ceros	No es un número (*Not a Number*, **NaN**)
0 ó 1	Todo ceros (00...0)	Todo ceros (00...0)	Representa al cero (0)
0 ó 1	Todo ceros (00...0)	Distinta de todo ceros	Número muy pequeño cercano al cero (0)

Figura 3.5: tabla de condiciones especiales en el estándar IEEE 754

Ejemplo 3.6: utilizando los casos especiales de estándar IEEE 754, determine el valor decimal de la tabla

Signo (S)	Exponente €	Mantisa (m)	Valor decimal
0	255	0	**+∞**
1	128	10000001100001010101011	-4244139=-4.244139x10^6
1	1022	000000000000000000000000000000 0000000000000010101011	-171x2^{-1}=85.5=8.55x10^{-1}
0	2047	10000001111…101010101111	**NaN**

171x2^{-1}=171*0.5=85.5=8.55x10^{-1}

PROBLEMAS RESUELTOS

Problema 1: exprese en el formato IEEE 754 precisión simple el número 5777
N = 5777 = 1011010010001_2 =1. 011010010001 x 2^{12}
Cuya representación será: (-1) ^{signo} x 1.mantisa x $2^{exponente - 127}$
S = 0
Exponente externo=12
Exponente en exceso=2^{8-1} -1=2^{7-}-1=127
E = exponente externo + exponente a exceso=12+127 = 139 = 10001011_2
m = 011010010001
$N_{(IEEE)}$ = 0 10001011 01101001000100000000000

Problema 2: exprese en el formato IEEE 754 precisión simple el número –0.75
Solución:
El número –0.75 = -3/4 = -3/2^2 = -11_2 / 2^2 = - 0.11
En notación científica normalizada se tendría: - 1.1 x 2^{-1}
Exponente externo=-1; exponente a exceso en precisión simple=127
De tal manera: exponente E= 127 + (–1) = 126 =01111110_2
Comparando con la expresión: (-1) ^{signo} x 1.mantisa x $2^{exponente - 127}$
Por lo tanto, la representación del –0.75 es:

31	30 . . . 23	22 . . . 0
1	0 1 1 1 1 1 1 0	10000000000000000000000
signo	Exponente	Mantisa

Problema 3: obtenga el número decimal con 6 cifras significativas los números asociados en precisión simple,

1) $N_{(IEEE)}$ = 1 10000001 01001001110001000000000
   Cuya representación es: (-1) ^{signo} x 1.mantisa x $2^{exponente - 127}$
   S = 1
   Exponente externo = 10000001_2 = 129 – 127 = 2
   m = 01001001110001000000000 => según la tabla de precisión simple agregue 1 a la mantisa:
   1.01001001110001000000000=101001001110001x2^{-14}
   N = (-1)x(101001001110001x2^{-14}) x 2^2 =-1.28814697265625x2^2=**-5**.152587890625=**-5.15259**

2) $N_{(IEEE)}$ = 0 11001100 10000111110100100000000
   S = 0
   E = 11001100_2 = 204 – 127 = 77
   m = 10000111110100100000000
   Agregue el bit 1 a la mantisa, como lo indica la primera fila de la tabla de precisión simple:
   N_2 =1.100001111101001_2.
   Escriba el número sin normalizar (moviendo en punto a la derecha):
   N_2 =1100001111101_2x2^{-15}
   Escriba el número en base 10 según la tabla de precisión doble:
   N = (+1) x (1100001111101_2*2^{-15}) x 2^{77}
   = (+1)x(0.191314697265625)x2^{77}

= 28.910.659.649.521.294.770.176
=$2.8910659649521294770176x10^{22}$
=$\mathbf{2.89107x10^{22}}$

Problema 5: exprese en decimal el número $805C0000_H$ del estándar IEEE 754 con precisión simple y a qué número representa en base 10, con 8 cifras significativas.
Procedamos como sigue para obtener la solución:
Convierta $805C0000_H$ a base 2:

8 0 5 C 0 0 0 0

1000 0000 0101 1100 0000 0000 0000 0000

1 00000000 10111000000000000000000
Obtenga los bits del signo, de la mantisa y del exponente:

31	30 . . . 23	22 . . . 0
1	00000000	10111000000000000000000
signo	Exponente	Mantisa

Como puede observar, todos los bits del exponente son ceros (00000000) y la mantisa es distinta de todo ceros, se deduce que es un caso especial: $(-1)^{signo}$ x 0.mantisa x 2^{-126}.
Efectivamente, se está representado a un número muy pequeño sin bit implícito con exponente -126.
En notación exponencial, puesto que en este caso no se utiliza bit implícito, la mantisa se escribe con un cero (0), seguido de la coma decimal (,) y de los bits de la mantisa (10111000000000000000000). En cuanto al signo del número puede decirse que es negativo, y se representa con el bit de signo 1. Con todo ello, el número es:
$-0,10111 \times 2^{-126}$
Para expresar el número en base 10: Signo S=1; el exponente $E=00000000_2=0$. La mantisa $m=10111000000000000000000_2$. Por lo tanto, utilizando la última fila de la tabla de precisión simple se antepuso el bit 0.

$-0.10111x2^{-126}=-(2^{-1}+2^{-3}+2^{-4}+2^{-5})x2^{-126}$
$=-(0.5+0.125+0.0625+0.03125)x\ x2^{-126}$
$=-0.71875x1.1754943508222875079687365372222x10^{-38}$
$=-8.4488656465351914635252938612849x10^{-39}$
$=\mathbf{-8.4488656\ x10^{-39}}$

Problema 6: exprese en decimal el número $1BACAFEA_H$ del estándar IEEE 754 con precisión simple con 6 dígitos de precisión
Solución:
$1BACAFEA_H=000110111\ 01011001010111111101010_2$

31	30 . . . 23	22 . . . 0
0	00110111	01011001010111111101010
Sign o	Exponente	Mantisa

S=0 (positivo); $E=00110111_2=55$;

m=1.01011001010111111101010=101011001010111111101010x2^{-23}=11.317.226x2^{-23}

➔m=1,3491184711456298828125

Signo=$(-1)^s$*1=$(-1)^0$*1=1

m*2^{E-127}= 1,3491184711456298828125x2^{55-127}=1,3491184711456298828125x2^{-72}

= 2,8568694870242465761640176716675x10^{-22}

=+2,85687x10^{-22} con 6 dígitos de precisión

Problema 7: exprese en decimal el número 80D3EE45$_H$ del estándar IEEE 754 con precisión simple con 6 dígitos de precisión.

Solución:

80D3EE45$_H$ =10000000110100111110111001000101$_2$

31	30 . . . 23	22 . . . 0
1	00000001	10100111110111001000101
Signo	Exponente	Mantisa

S=1 (negativo); E=00000001$_2$=1;

m=1.10100111110111001000101=110100111110111001000101x2^{23}=13.889.093x2^{23}

Signo=$(-1)^s$*1=$(-1)^1$*1=-1

$(-1)^1$*1*m*2^{E-127}

= -(13.889.093x2^{23})*2^{1-127}

=1,3695703095848523149326751667461x10^{-24}

=-1,36957x10^{-24}

Problema 8: calcule el valor decimal con 6 cifras significativas a partir de los siguientes números:

2. **N_{IEEE}= 00011101100110110000000000000000$_2$**
   De acá se obtiene que S=0; E=00111011$_2$=59; m=00110110000000000000000$_2$, de donde se obtiene e=59-127=-68. Por lo tanto, utilice la fila 1 de la tabla:
   $(-1)^0$*1.0011011$_2$*2^{-68}= +(1+2^{-3}+2^{-4}+2^{-6}+2^{-7})*2^{-68}=1.2109375*2^{-68}=4.10281538263*10^{-21}
   =**+4.10281*10^{-21}**

3. **N_{IEEE}= 01111111110000000000000000000000$_2$**
   S=0; E=11111111$_2$=255; m=00000000000000000000000$_2$**=**0. Por lo tanto, **NaN**

4. **N_{IEEE}= 11111111100000000000000000000000$_2$**
   De acá se obtiene que S=1; E=11111111$_2$=255; m=00000000000000000000000$_2$=0. Por lo tanto, **N=-∞**

5. **N_2= 0.000000111111111000000000000$_2$**
   Normalicemos el número: **N_2**= 1.11111111000000000000$_2$*2^{-7}. Elimínese el bit implícito y se obtiene la mantisa completando con 0 los bits faltantes:
   S=0; E=-7; m= 11111111000000000000000$_2$
   Utilice la fila 1: $(-1)^0$*1. 11111111000000000000000$_2$*2^{-7}=+1.5594482421875*10^{-2}
   =**+1.55945*10^{-2}**

6. N_2= 10010011.101$_2$

Normalicemos el número: **N_2**= $1.0010011101_2 * 2^7$. Elimínese el bit implícito y se obtiene la mantisa completando con 0 los bits faltantes:
S=0; E=7; m= $00100111010000000000000_2$
Utilice la fila 1: $(-1)^0 * 1.00100111010000000000000_2 * 2^7 = +1.1533203125 * 2^7$ =**$+1.47625*10^2$**

7. N_{IEEE}= $10000000000000000000000000000000_2$

De acá se obtiene que S=1; E=00000000_2=0; m=$00000000000000000000000_2$=0. Por lo tanto, **N=0**

8. N_{IEEE}= $10000000010101000000000000000000_2$

De acá se obtiene que S=1; E=00000000_2=0; m=$10101000000000000000000_2 \neq 0$. Por consiguiente,
Utilice la fila 1: $(-1)^1 * 0.10101000000000000000000_2 = -(0+2^{-1}+2^{-3}+2^{-5}) * 2^{-126}$
$= -0.65625 * 2^{-126}$=**$-7.71418*10^{-39}$**

Problema 9: obtenga el número representado en IEEE-754 y el correspondiente expresado en hexadecimal, a partir de los siguientes números:

1. **N_2=$101110.010101110100001111100001111100010011_2$**
   Normalícese: $N_2 = 1.01110010101110100001111100001111100010011_2 * 2^5$
   De N_2 puede deducirse que e=5; E=5+127=132=10000100_2; S=0;
   m=$01110010101110100001111\cancel{100001111100010011}_2$
   Por lo tanto, N_{IEEE}=$0100\ 0010\ 0011\ 1001\ 0101\ 1101\ 0001\ 0000_2$
   N_H=$42395D10_H$

2. **N=37.296875**

Dicho número expresado en binario es:
$N_2 = 100101.010011_2 = 1.00101010011_2 * 2^5$
Ahora, obtengamos la mantisa eliminando el bits implícito del binario anterior, m=00101010011; s=0; e=5; por lo tanto, E=5+127=132=10000100. Por lo tanto,
N_{IEEE}=$01000010000101010011000000000000_2$
N_H=42153000_H

Problema 10: utilice el algoritmo en punto flotante para sumar los números 0.5 y –0.4375, suponiendo 4 bits de precisión y escriba la respuesta en decimal con 4 cifras de precisión.
Solución:
Primero obtendremos la versión binaria normalizada de cada número, suponiendo 4 bits de precisión:
$0.5 = 1/2 = 1/2^1 = 0.1 = 0.1 \times 2^0 = 1.000 \times 2^{-1}$
$- 0.4375 = - 7/16 = - 7/2^4 = - 0.0111 = 0.0111 \times 2^0 = - 1.110 \times 2^{-2}$
Ahora siguiendo el algoritmo de la suma:
1) Desplace al número con exponente más pequeño (a la derecha), hasta alinearlo con el exponente mayor: $- 1.110 \times 2^{-2} = - 0.111 \times 2^{-1}$
2) Sume las mantisas: $(1.000_2 \times 2^{-1}) + (- 0.111_2 \times 2^{-1}) = 0.001_2 \times 2^{-1}$
3) Normalice la suma, verificando si existe sobre flujo o bajo flujo:
$0.001_2 \times 2^{-1} = 0.010_2 \times 2^{-2} = 0.100_2 \times 2^{-3} = 1.000_2 \times 2^{-4}$

Puesto que 127 > - 4 > - 126, no hay sobre flujo ni bajo flujo. El exponente E= (– 4) + 127 = 123, y está entre 1 y 254).
4) Redondee la suma: $1.000_2 \times 2^{-4}$
Por lo tanto, el resultado es: $\mathbf{1.000_2 \times 2^{-4}} = 0.0001000_2 = 1/2^4 = 1/16 =$ **0.0625=6.250x10^{-2}**

Problema 11: utilice el algoritmo en punto flotante para multiplicar los números 0.5 por - 0.4375 en binario y exprese el resultado en decimal con 4 cifras de precisión y en binario con 4 bits de precisión.
Solución:
Primero obtendremos la versión binaria normalizada, suponiendo 4 bits de precisión en cada número: $0.5=0.1_2= 1.000_2 \times 2^{-1}$ y $-0.4375= -0.0111_2= -1.110_2 \times 2^{-2}$.
Siguiendo el algoritmo de la multiplicación se tiene:
1) Sume los exponentes: (-1) + (-2) = -3
2) Multiplique las mantisas: $1.000_2 \times 1.110_2 =1110000_2$. El producto es $1.110000_2 \times 2^{-3}$, pero necesitamos mantenerlo en 4 bits por lo que obtenemos: $1.110_2 \times 2^{-3}$
3) Verifique si el producto está normalizado y que no haya error de sobre flujo (overflow) o bajo flujo (underflow). Efectivamente, está normalizado y no existe error de flujo.
4) Redondee el producto. Desde luego, no cambia al producto.
5) Como los signos de los operandos son diferentes, el resultado es: $\mathbf{-1.110_2 \times 2^{-3}= 0,21875}$

Problema 12: si a= 0.4523×10^4 y b = 0.0002115×10^0, calcule a + b y a – b, almacenando solamente 4 dígitos en la mantisa y 2 en el exponente.
Solución:
a + b = $0.4523\times10^4 + 0.0002115\times10^0$
= $0.4523 \times10^4 + 0.00000002115 \times10^4 = 0.4523 \times10^4$ positivo=$\mathbf{+4.523\times10^3}$.
Ahora, a - b = 0.4523×10^4 positivo=$\mathbf{+4.523\times10^3}$

AUTOEVALUACION 3

1. Obtenga el número decimal con **4** cifras significativas, asociado al número dado en IEEE 754: $11000000101001001110001000000000_2$
   A) -5.153 B) -3.288 C) -5.152 D) -3.289

2) Obtenga el número decimal con 5 dígitos de precisión en la mantisa y 2 en el exponente, asociado al número dado en IEEE 754: $11000000101001001110001000000000_2$
   A) 5.15259 B) 5.15258 C) -5.15258 D) -5.15259

3) Exprese en el formato IEEE 754 el número 57.23 en precisión simple.
   A) 0 10000100 11001001110000000000000
   B) 0 10000101 11001001110000000000000
   C) 1 10000100 11001001110000000000000
   D) 1 10000101 11001001110000000000000

TALLER 3

1. Normalice los siguientes números y expréselos con 4 cifras significativas:
   32452.25661x10^{-3}= ____________________________
   932032.00001233x10^{4}= ____________________________
   0.000001234434x10^{7}= ____________________________
   0.000001234434x10^{-5}= ____________________________
2. Represente a –0.75 en punto flotante de precisión simple expresado en formato IEEE 754.
   R/

31	30 . . . 23	22 . . . 0
1	0 1 1 1 1 1 1 0	10000000000000000000000
signo	Exponente	Mantisa

3. A partir de los datos en la siguiente tabla, complétela determinando el valor decimal, según las codificaciones con significado especial de la IEEE 754:

Signo (S)	**Exponente (E)**	**Mantisa (m)**	**Valor decimal**
1	255	0	
1	247	10000001111101010101111	
0	137	10000001111101010101111	
0	2047	010101011…11100111001111	
0	255	0	
0	1250	10000001111…101010101111	
1	2047	10000001111…101010101111	
1	2047	0	
1	255	0	

La siguiente tabla, de potencias de 2 es útil para las conversiones decimal-binario-decimal

POTENCIAS DE 2

-1	-2	-3	-4	-5	-6	-7
0.5	0.25	0.125	0.0625	0.03125	0.015625	0.078125
-8	-9	-10	-11	-12	-13	-14
0,00390625	0,001953125	0,0009765625	0,00048828125	0,000244140625	0,0001220703125	0,00006103515625