

1) y 2) están en un ppt aparte.

3) a)

$$N_1 = 0 \text{ (el más chico)}$$

$$N_2 = \underbrace{0,11111111}_{8 \text{ bits}} \times 2^{1111} = (1 - 2^{-8}) \times 2^{15} = (2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-5} + 2^{-6} + 2^{-7} + 2^{-8}) \times 2^{15} \text{ (el más grande)}$$

$$N_3 = 0,00000001 \times 2^{0000} = 2^{-8} \text{ (el más cercano a } N_1)$$

$$N_4 = 0,11111110 \times 2^{1111} \text{ (el más cercano a } N_2)$$



$$\text{Rango} = [N_1; N_2] = [0; (1 - 2^{-8}) \times 2^{15}]$$

$$R_1 = \text{Resolución extremo inferior} = N_3 - N_1 = (0,00000001 - 0) \times 2^{0000} = 2^{-8}$$



$$R_2 = \text{Resolución extremo superior} = N_2 - N_4 = (0,11111111 - 0,11111110) \times 2^{15} = 2^{-8} \cdot 2^{15} = 2^7$$



b)

$$N_1 = 0, \underbrace{1000000000000000}_{15 \text{ bits}} \times 2^{\overbrace{1000000000}^{10 \text{ bits}}} = 2^{-1} \times 2^{-511} = 0,5 \times 2^{-511}$$

$$N_2 = 0,1111111111111111 \times 2^{0111111111} = (1 - 2^{-15}) \times 2^{+511}$$

$$N_4 = 0,1111111111111110 \times 2^{0111111111}$$

$$N_3 = 0,1000000000000001 \times 2^{1000000000} = (2^{-1} + 2^{-15}) \times 2^{-511}$$

$$R_1 = \text{Resolución extremo inferior} = N_3 - N_1 = (0,1000000000000001 - 0,1000000000000000) \times 2^{1000000000} = 0,0000000000000001 \times 2^{1000000000} = 2^{-15} \cdot 2^{-511}$$



$$R_2 = \text{Resolución extremo superior} = N_2 - N_4 = (0,1111111111111111 - 0,1111111111111110) \times 2^{0111111111} = 0,0000000000000001 \times 2^{0111111111} = 2^{-15} \times 2^{+511}$$



Organización de Computadoras –Fac. de Informática – Prof. Jorge M. Runco  
Práctica 3 – Punto Flotante – Curso 2020

$$\text{Rango} = [N_1; N_2] = [0,5 \times 2^{-511}; (1 - 2^{-15}) \times 2^{+511}]$$

Como la mantisa está normalizada el 0 (cero) y una zona “cercana” a él, no están incluidos en el rango.

c)

$$N_1 = 0 \underbrace{0,1000000000000000}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{00000} = + (2^{-1}) \times 2^{-16} = + 0,5 \times 2^{-16} \text{ (el más chico +)}$$

$$\text{Exponente en exceso} = 10000 + 10000 = 00000 \text{ (-16 en exceso)}$$

$$N_2 = 0 \underbrace{0,111111111111111}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{11111} = + (1 - 2^{-15}) \times 2^{+15} \text{ (el más grande +)}$$

$$\text{Exponente en exceso} = 01111 + 10000 = 11111 \text{ (+15 en exceso)}$$

$$N_3 = 0 \underbrace{0,1000000000000001}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{00000} = + (2^{-1} + 2^{-15}) \times 2^{-16} \text{ (el más cercano a } N_1)$$

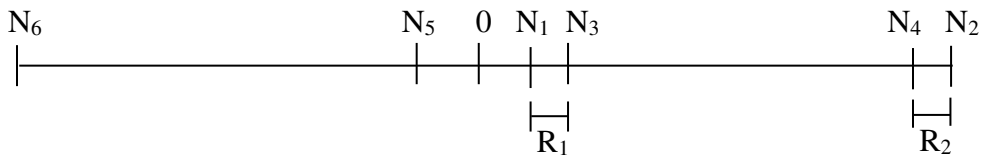
$$N_4 = 0 \underbrace{0,111111111111110}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{11111} \text{ (el más cercano a } N_2)$$

$$N_5 = 1 \underbrace{0,1000000000000000}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{00000} = - (2^{-1}) \times 2^{-16} = - 0,5 \times 2^{-16}$$

$$N_6 = 1 \underbrace{0,111111111111111}_{15 \text{ bits} + 1 \text{ Signo} = 16} \times 2^{11111} = - (1 - 2^{-15}) \times 2^{+15}$$

$$R_1 = N_3 - N_1 = (2^{-1} + 2^{-15} - 2^{-1}) \times 2^{-16} = (2^{-15}) \times 2^{-16} = 2^{-31} \quad \leftarrow$$

$$R_2 = N_2 - N_4 = (0,111111111111111 - 0,111111111111110) \times 2^{+15} = 2^{-15} \times 2^{+15} = 2^0 = 1 \quad \leftarrow$$



$$\text{Rango} = [N_6; N_5] \cup [N_1; N_2] = [- (1 - 2^{-15}) \times 2^{+15}; - 0,5 \times 2^{-16}] \cup [+ 0,5 \times 2^{-16}; + (1 - 2^{-15}) \times 2^{+15}]$$

d)

$$\text{Rango} = [- (1 - 2^{-N}) \times 2^{E1}; - 0,5 \times 2^{E2}] \cup [+ 0,5 \times 2^{E2}; + (1 - 2^{-15}) \times 2^{E1}]$$

$$E1 = + (2^{M-1} - 1) \quad E2 = - (2^{M-1})$$

4)

0

Sin normalizar

$0 = 0,000000000 \times 2^E \rightarrow E$  puede ser cualquier exponente.

Normalizada

$0 =$  No se puede representar. El cero no está en el rango.

Normalizada con bit implícito

$0 =$  No se puede representar. El cero no está en el rango.

---

+ 1

Sin normalizar

$1 = 0,100000000 \times 2^{00001} = 0,010000000 \times 2^{00010} = 0,001000000 \times 2^{00011} = \dots\dots\dots$

9 bits +1 Signo = 10

0	00011	001000000
---	-------	-----------

Normalizada

$1 = 0,100000000 \times 2^{00001}$

9 bits +1 Signo = 10

0	00001	100000000
---	-------	-----------

Normalizada con bit implícito

$1 = 0,100000000 \times 2^{00001}$

10 bits +1 Signo = 11

0	00001	000000000
---	-------	-----------

---

+ 9

Sin normalizar

$9 = 0,100100000 \times 2^{00100} = + (2^{-1} + 2^{-4}) \times 2^4 = + (0,5 + 0,0625) \times 16 = 9$   
 $9 = 0,010010000 \times 2^{00101} = + (2^{-2} + 2^{-5}) \times 2^5 = + (0,25 + 0,03125) \times 32 = 9$   
 $9 = 0,001001000 \times 2^{00110} = + (2^{-3} + 2^{-6}) \times 2^6 = + (0,125 + 0,015625) \times 64 = 9$

0	00110	001001000
---	-------	-----------

Normalizada

$9 = 0,100100000 \times 2^{00100} = + (2^{-1} + 2^{-4}) \times 2^4 = + (0,5 + 0,0625) \times 16 = 9$

0	00100	100100000
---	-------	-----------

Normalizada con bit implícito

$9 = 0,100100000 \times 2^{00100} = + (2^{-1} + 2^{-4}) \times 2^4 = + (0,5 + 0,0625) \times 16 = 9$

0	00100	001000000
---	-------	-----------

- 5,0625

Sin normalizar

$$\begin{aligned} -5,0625 &= 1 \ 101,000100 \times 2^0 = 1 \ 0,101000100 \times 2^{00011} = -(2^{-1}+2^{-3}+2^{-7}) \times 2^3 = -0,6328125 \times 8 = -5,0625 \\ -5,0625 &= 1 \ 0,010100010 \times 2^{00100} = -(2^{-2}+2^{-4}+2^{-8}) \times 2^4 = -0,31640625 \times 16 = -5,0625 \\ -5,0625 &= 1 \ 0,001010001 \times 2^{00101} = -(2^{-3}+2^{-5}+2^{-9}) \times 2^5 = -0,158203125 \times 32 = -5,0625 \end{aligned}$$

1	00101	001010001
---	-------	-----------



Normalizada

$$-5,0625 = 1 \ 0,101000100 \times 2^{00011} = -(2^{-1}+2^{-3}+2^{-7}) \times 2^3 = -0,6328125 \times 8 = -5,0625$$

1	00011	101000100
---	-------	-----------

Normalizada con bit implícito

$$-5,0625 = 1 \ 0,1010001000 \times 2^{00011} = -(2^{-1}+2^{-3}+2^{-7}) \times 2^3 = -0,6328125 \times 8 = -5,0625$$

1	00011	010001000
---	-------	-----------

---

+ 34000,5

Sin normalizar

$$\begin{aligned} \text{El más grande} &= 0 \ 0,111111111 \times 2^{01111} = +(1 - 2^{-9}) \times 2^{15} = +(0,998046875) \times 32768 = +32704 \\ \text{Por lo tanto } 34000,5 &\text{ está fuera de rango.} \end{aligned}$$

Normalizada

$$\begin{aligned} \text{El más grande} &= 0 \ 0,111111111 \times 2^{01111} = +(1 - 2^{-9}) \times 2^{15} = +(0,998046875) \times 32768 = +32704 \\ \text{Por lo tanto } 34000,5 &\text{ está fuera de rango.} \end{aligned}$$

Normalizada con bit implícito

$$\begin{aligned} \text{El más grande} &= 0 \ 0,1111111111 \times 2^{01111} = +(1 - 2^{-10}) \times 2^{15} = +(0,998046875) \times 32768 = +32736 \\ \text{Por lo tanto } 34000,5 &\text{ está fuera de rango.} \end{aligned}$$

---

+ 0,015625

$$0,015625 \times 2 = 0,03125$$

$$0,03125 \times 2 = 0,0625$$

$$0,0625 \times 2 = 0,125$$

$$0,125 \times 2 = 0,25$$

$$0,25 \times 2 = 0,5$$

$$0,5 \times 2 = 1,0$$

$$0,0 \times 2 = 0,0$$

Sin normalizar

$$+ 0,015625 = 0\ 0,000001000 \times 2^0 = + (2^{-6}) \times 2^0 = + 0,015625 \times 1 = + 0,015625$$

$$+ 0,015625 = 0\ 0,000010000 \times 2^{-1} = + (2^{-5}) \times 2^{-1} = + 0,03125 \times 0,5 = + 0,015625$$

$$+ 0,015625 = 0\ 0,000100000 \times 2^{-2} = + (2^{-4}) \times 2^{-2} = + 0,0625 \times 0,25 = + 0,015625$$

0	11110	000100000
---	-------	-----------



Normalizada

$$+ 0,015625 = 0\ 0,100000000 \times 2^{-5} = 0\ 0,100000000 \times 2^{11011} = + 0,5 \times 0,03125 = + 0,015625$$

0	11011	100000000
---	-------	-----------

Normalizada con bit implícito

$$+ 0,015625 = 0\ 0,100000000 \times 2^{-5} = 0\ 0,100000000 \times 2^{11011}$$

0	11011	000000000
---	-------	-----------

## Número máximo

Sin normalizar

$$\text{El más grande} = 0\ 0,111111111 \times 2^{01111} = + (1 - 2^{-9}) \times 2^{15} = + (0,998046875) \times 32768 = + 32704$$

0	01111	111111111
---	-------	-----------

Normalizada

$$\text{El más grande} = 0\ 0,111111111 \times 2^{01111} = + (1 - 2^{-9}) \times 2^{15} = + (0,998046875) \times 32768 = + 32704$$

0	01111	111111111
---	-------	-----------

Normalizada con bit implícito

$$\text{El más grande} = 0\ 0,111111111 \times 2^{01111} = + (1 - 2^{-10}) \times 2^{15} = + (0,998046875) \times 32768 = + 32736$$

0	01111	111111111
---	-------	-----------

## Número mínimo

Organización de Computadoras –Fac. de Informática – Prof. Jorge M. Runco  
Práctica 3 – Punto Flotante – Curso 2020

Sin normalizar

El más chico =  $1\ 0,111111111 \times 2^{01111} = - (1 - 2^{-9}) \times 2^{15} = - (0,998046875) \times 32768 = - 32704$

1	01111	111111111
---	-------	-----------

Normalizada


El más chico =  $1\ 0,111111111 \times 2^{01111} = - (1 - 2^{-9}) \times 2^{15} = - (0,998046875) \times 32768 = - 32704$


1	01111	111111111
---	-------	-----------


Normalizada con bit implícito


El más chico =  $1\ 0,111111111 \times 2^{01111} = + (1 - 2^{-10}) \times 2^{15} = + (0,998046875) \times 32768 = + 32736$


1	01111	111111111
---	-------	-----------


6)  $00001111 \ 00000011 + 00001000 \ 00000010 = 15 \times 2^3 + 8 \times 2^2 = 120 + 32 = 152$  

$$\begin{array}{r}
 + \quad 00001111 \times 2^3 \\
 \quad 00000100 \times 2^3 \\
 \hline
 \quad 00010011 \times 2^3
 \end{array}
 \begin{array}{l}
 \xrightarrow{\text{red arrow}} 15 \times 2^3 \\
 \xrightarrow{\text{red arrow}} 4 \times 2^3 \\
 \\
 \end{array}
 = 19 \times 2^3 = 19 \times 8 = 152$$


$01111111 \ 00000000 + 11111100 \ 10000001 = 127 \times 2^0 + 252 \times 2^{-1} = 127 + 126 = 253$  

$$\begin{array}{r}
 \quad 111111 \\
 + \quad 01111111 \times 2^0 \\
 \quad 01111110 \times 2^0 \\
 \hline
 \quad 11111101 \times 2^0
 \end{array}
 \begin{array}{l}
 \xrightarrow{\text{red arrow}} 127 \times 2^0 \\
 \xrightarrow{\text{red arrow}} 126 \times 2^0 \\
 \\
 \end{array}
 = 253 \times 2^0 = 253$$


$00000001 \ 00000111 + 00011100 \ 00000000 = 1 \times 2^7 + 28 \times 2^0 = 128 + 28 = 156$  


$$\begin{array}{r}
 \quad 10000000 \times 2^0 \\
 \quad 00011100 \times 2^0 \\
 \hline
 \quad 10011100 \times 2^0
 \end{array}
 \begin{array}{l}
 \xrightarrow{\text{red arrow}} 128 \times 2^0 \\
 \xrightarrow{\text{red arrow}} 28 \times 2^0 \\
 \\
 \end{array}
 = 156 \times 2^0 = 156$$


7)

8,625


a) Mantisa fraccionaria normalizada de 5 bits BSS, exponente Ca2 4 bits.

$8,625 = 1000,101 \times 2^0 = 0,1000101 \times 2^4$  Mantisa sólo con 5 bits,   $0,10001 \times 2^4$

$= (2^{-1} + 2^{-5}) \times 2^4 = 8,5$  

0100	10001
------	-------

El N° que le sigue  $= (0,10001 + 0,00001) \times 2^4 = 0,10010 \times 2^4 = (2^{-1} + 2^{-4}) \times 2^4 = 9$ .

$\left. \begin{array}{l} \text{Error1} = 8,625 - 8,5 = 0,125 \\ \text{Error2} = 9 - 8,5 = 0,5 \end{array} \right\}$ 
 Menor error (valor más cercano) es 8,5 

8,625 no tiene una representación exacta en este sistema. 8,5 está más cerca que 9.

b) Mantisa fraccionaria normalizada de 10 bits BCS, exponente 3 bits Ca2.

$8,625 = 0 \ 1000,101 \times 2^0 = 0 \ 0,100010100 \times 2^4$

El exponente 4 no se puede expresar con 3 bits en Ca2

El número más grande =  $0,111111111 \times 2^{+3} = + (1 - 2^{-9}) \times 8 = (0,998046875) \times 8 = 7,984375$   
Éste sería el más cercano a 8,625.



0	011	111111111
---	-----	-----------

2,5

a)  $2,5 = 10,1 \times 2^0 = 0,101 \times 2^2 = 0,10100 \times 2^{0010} = (2^{-1} + 2^{-3}) \times 2^2 = (0,5 + 0,125) \times 4 = 2,5$



0010	10100
------	-------

b)  $2,5 = 0,101000000 \times 2^{010} = (2^{-1} + 2^{-3}) \times 2^2 = (0,5 + 0,125) \times 4 = 2,5$



0	010	101000000
---	-----	-----------

0,4

a)  $0,4 \times 2 = 0,8$        $0,4 \rightarrow 0,01100 \times 2^0$  corriendo la coma entra un dígito más  
 $0,8 \times 2 = 1,6$        $0,11001 \times 2^{-1} = 0,11001 \times 2^{1111} = 0,390625$

$0,6 \times 2 = 1,2$

$0,2 \times 2 = 0,4$

$0,4 \times 2 = 0,8$

$0,8 \times 2 = 1,6$

El que sigue =  $0,11010 \times 2^{-1} = (0,5 + 0,25 + 0,0625) \times 0,5 = 0,40625$

Esta representación es más cercana a 0,4.



Error1 =  $0,4 - 0,390625 = 0,009375$   
 Error2 =  $0,40625 - 0,4 = 0,00625$



b)  $0,4 \rightarrow 0,011001100 \times 2^0$

Normalizada =  $0,110011001 \times 2^{-1} = (2^{-1} + 2^{-2} + 2^{-5} + 2^{-6} + 2^{-9}) \times 0,5 = 0,399414062$

$0,4 \times 2 = 0,8$

$0,8 \times 2 = 1,6$

$0,6 \times 2 = 1,2$

$0,2 \times 2 = 0,4$

$0,4 \times 2 = 0,8$

$0,8 \times 2 = 1,6$


$0,6 \times 2 = 1,2$

$0,2 \times 2 = 0,4$

$0,4 \times 2 = 0,8$

$0,8 \times 2 = 1,6$



El que sigue =  $0,110011010 \times 2^{-1} = (2^{-1} + 2^{-2} + 2^{-5} + 2^{-6} + 2^{-8}) \times 0,5 = 0,400390625$  

$$\text{Error1} = 0,4 - 0,399414062 = 0,000585938$$

$$\text{Error2} = 0,400390625 - 0,4 = 0,000390625$$



0,400390625 está más cerca de 0,4 (menor error)

---

8) **8,625**

a)  $8,5 < 8,625 < (2^{-1} + 2^{-5}) \times 2^4 = 9$

$$E_A = 8,625 - 8,5 = 0,125 \quad \leftarrow \text{Menor error}$$

$$E_A = 9 - 8,625 = 0,375$$

$$E_R = E_A / N^{\circ} \text{ a representar} = 0,125 / 8,625 \sim 0,0145 \quad \leftarrow$$

b)  $E_A = 0$  Representación exacta

---

**2,5**

a)  $E_A = 0$  Representación exacta.

b)  $E_A = 0$  Representación exacta.

---

**0,4**

a)  $0,390625 < 0,4 < 0,40625$

$$E_A = 0,4 - 0,390625 = 0,009375$$

$$E_A = 0,40625 - 0,4 = 0,00625 \quad \leftarrow \text{Menor error}$$

$$E_R = 0,00625 / 0,4 = 0,015625 \quad \leftarrow$$

---

b)

$$E_A = 0,4 - 0,399414062 = 0,000585938$$

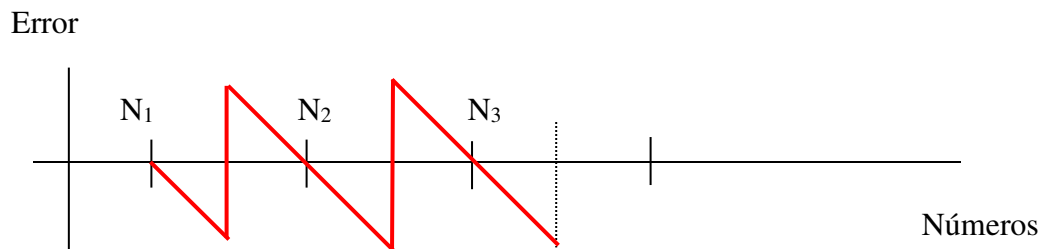
$$E_A = 0,400390625 - 0,4 = 0,000390625 \quad \leftarrow \text{Menor error}$$

$$E_R = 0,000390625 / 0,4$$

---

Organización de Computadoras –Fac. de Informática – Prof. Jorge M. Runco  
Práctica 3 – Punto Flotante – Curso 2020

10)



12)

$$0 \ 11000100 \ 000000000000000000000000 = +1,0 \ 2^{+69}$$

$$E = 196 - 127 = 69$$

$$1 \ 11111110 \ 101000000000000000000000 = -1,625 \times 2^{+127}$$

$$E = 254 - 127 = 127$$

$$0 \ 00000000 \ 000000000000000000000001 = +(2^{-23}) \times 2^{-126}$$

$$1 \ 00000000 \ 000000000000000000000000 = -0$$

$$0 \ 11111111 \ 000000000000000000000000 = +\infty$$

$$0 \ 11111111 \ 000001000000000000000000 = \text{NaN}$$

13)

$$0,0625 = 0 \ 0,0001000000 \dots 00 \times 2^0 = 0 \ 1,00000 \dots 000 \times 2^4 = 0 \ 1,000000 \dots 000 \times 2^{10000011}$$

$$E = 4 + 127 = 131 = 10000011 = (+4 \text{ en exceso } 127)$$

0	10000011	00000000.....0000
---	----------	-------------------

$$-40000 = 1 \ 1001110001000000 \times 2^0 = 1 \ 1,001110001000000000000000 \times 2^{15} =$$

$$= 1 \ 1,001110001000000000000000 \times 2^{10001110}$$

0	10001110	00111000100000.....0000
---	----------	-------------------------

$$+1 = 0 \ 1,00000 \dots 000 \times 2^0 = 0 \ 1,00000 \dots 000 \times 2^{01111111}$$

$$E = 0 + 127 = 127 = 01111111 = (0 \text{ en exceso } 127)$$

0	01111111	00000000000000.....0000
---	----------	-------------------------

Organización de Computadoras –Fac. de Informática – Prof. Jorge M. Runco  
Práctica 3 – Punto Flotante – Curso 2020

+ 13 = 0 1101,00000...000 x  $2^0 = 0 1,101000...000$  x  $2^{+3} = 0 1,101000...000$  x  $2^{10000010}$   
E = 3 + 127 = 130 = 10000010 (+ 3 en exceso 127)

0	10000010	10100000000000.....0000
---	----------	-------------------------

---