

Computación Numérica

Primer Parcial A - Febrero 2014

1. Si el número

<i>signo</i>	1 bit	1
<i>exponente</i>	8 bits	10001110
<i>mantisa</i>	23 bits	10101100000000000000000

sigue la norma IEEE 754 para representación en punto flotante con precisión simple, calcular su representación en base 10.

Cálculo del exponente

El exponente sesgado es $2^7 + 2^3 + 2^2 + 2^1 = 142$. Si el número de bits del exponente es $m = 8$, entonces el *sesgo* = $2^{m-1} - 1 = 127$ y el exponente es

$$142 - \text{sesgo} = 142 - 127 = 15$$

Cálculo de la mantisa

Teniendo en cuenta el bit escondido, la mantisa es (1).101011 que en base 10 es

$$1 + 2^{-1} + 2^{-3} + 2^{-5} + 2^{-6} = 1.671875$$

Número

Como el bit del signo es 1 el número es negativo. Teniendo en cuenta la mantisa y el exponente, el número en base 10 es

$$-(1 + 2^{-1} + 2^{-3} + 2^{-5} + 2^{-6})2^{15} = -54784$$

2. Redondear más cercano par, si la precisión es 4, los siguientes números en base 2:

$n_1 = 0,110010 \rightarrow 0,1100$ es equidistante de dos números y se redondea al más cercano par que es el más cercano por debajo.

$n_2 = 1,111100 \rightarrow 10,00$ es equidistante de dos números y se redondea al más cercano par que es el más cercano por arriba.

$n_3 = 1,010110 \rightarrow 1,011$ se redondea hacia arriba porque la parte a truncar es mayor que 0,0001.

$n_4 = 1,010010 \rightarrow 1,010$ se redondea hacia abajo porque la parte a truncar es menor que 0,0001.

3. Una máquina almacena números en punto flotante en 12 bits. El primer bit se usa para el signo del número, los cinco siguientes para el exponente sesgado y los últimos seis bits para la magnitud de la mantisa. Si se sigue un criterio similar al de la norma IEEE 754:

- (a) ¿Cuál sería el ϵ de máquina expresado en base 10 y con este formato?

Formato decimal

El número 1 teniendo en cuenta el bit escondido sería $(1).000000 \times 2^0$. El ϵ de máquina es el número más pequeño que se puede alinear con este 1 en este formato para sumárselo:

$$\begin{cases} 1 = 1.000000 \times 2^0 \\ \epsilon = 0.000001 \times 2^0 = 1.000000 \times 2^{-6} = 2^{-6} = 0.015625 \end{cases}$$

Formato binario

En este caso $m = 5$, entonces el *sesgo* $= 2^{m-1} - 1 = 15$ y el exponente sesgado será $-6 + \text{sesgo} = 9 = 2^3 + 1$ y que en binario será 1001. Por lo tanto, teniendo en cuenta la mantisa, $(1).000000$, en este formato binario

$$\epsilon = 0\ 01001\ 000000$$

- (b) ¿Cuál es el mayor real positivo que se almacena en forma desnormalizada?

Formato decimal

Si se almacena en forma desnormalizada:

- El exponente en binario es 00000.
- Se asume como exponente el menor posible de los normalizados.
- Se asume que el bit escondido es 0.

El menor exponente posible es, sesgado, 00001 y sin sesgar $1 - \text{sesgo} = 1 - 15 = -14$. Por lo tanto, el mayor número desnormalizado es $(0).111111 \times 2^{-14}$ que en decimal es

$$(2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-5} + 2^{-6}) \times 2^{-14} = 0.000060081$$

Formato binario

Ya se ha dicho que el exponente se escribe 00000 y como la mantisa es $(0).111111$, teniendo en cuenta el signo, se escribirá

$$0\ 00000\ 111111$$

1. ¿Cómo se almacenaría en precisión simple según la norma IEEE 754 el número 215,90625?

En precisión simple tenemos 32 bits (4 bytes) en total: 1 bit para el signo, 8 bits para el exponente y 23 para la mantisa.

Cálculo de la mantisa

Para convertir a binario la parte decimal, multiplicamos por 2, le quitamos la parte entera, que será nuestro dígito binario y repetimos el proceso.

$$\begin{array}{rcccccc} \text{Decimal :} & 0.90625 & 0.8125 & 0.625 & 0.25 & 0.5 & 0 \\ \text{Entera :} & & 1 & 1 & 1 & 0 & 1 \end{array}$$

y tomamos los dígitos:

$$0.11101$$

Para convertir a binario la parte entera 118 dividimos de forma reiterada por 2 y guardamos los restos:

$$\begin{array}{rcccccccc} \text{cocientes :} & 215 & 107 & 53 & 26 & 13 & 6 & 3 & 1 \\ \text{restos :} & 1 & 1 & 1 & 0 & 1 & 0 & 1 & \end{array}$$

empezando por el último cociente, seguimos con los restos en orden inverso:

$$11010111$$

Para almacenar según la norma IEEE:

$$11010111.11101 = (1).10101111101 \times 2^7$$

y no hace falta almacenar el primer uno. Por lo tanto la mantisa se almacena como

$$101011111101000000000000$$

Cálculo del exponente

Si el número de bits del exponente es $m = 8$, entonces el *sesgo* = $2^{m-1} - 1 = 127$. El exponente sesgado será $\text{exponente} + \text{sesgo} = 7 + 127 = 134 = 2^7 + 2^2 + 2^1$, que en binario es 10000110.

Número	signo	exponente	mantisa
	0	10000110	101011111101000000000000

2. ¿Con cuantos dígitos significativos aproxima $x^* = 100$ a $x_2 = 99.99997$? Entonces, ¿cómo deberíamos escribir x^* ?

$$\left| \frac{100 - 99.99997}{99.99997} \right| = 3 \times 10^{-7} < 5 \times 10^{-7}$$

y el número de dígitos significativos es 7. Por lo tanto, deberíamos escribir $x^* = 100.0000$.

3. Una máquina almacena números en punto flotante en 13 bits. El primer bit se usa para el signo del número, los seis siguientes para el exponente sesgado y los últimos seis bits para la magnitud de la mantisa. Si se sigue un criterio similar al de la norma IEEE 754:

- (a) ¿Cuál es el mayor real positivo que se almacena en forma normalizada?

Formato decimal

El mayor exponente es igual al sesgo, en este caso $m = 6$, entonces el *sesgo* $= 2^{m-1} - 1 = 31$, que junto con la mayor mantisa posible nos da el número $(1).111111 \times 2^{31}$ que en base 10 es

$$(1 + 2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} + 2^{-5} + 2^{-6}) \times 2^{31} = 4261412864$$

Formato binario

El mayor exponente posible es 111110 por lo tanto, teniendo en cuenta la mantisa del apartado anterior, el número es

$$0 \ 111110 \ 111111$$

- (b) ¿Cuál es el menor real positivo que se almacena en forma desnormalizada?

Formato decimal

Si se almacena en forma desnormalizada:

- El exponente en binario es 00000.
- Se asume como exponente el menor posible de los normalizados.
- Se asume que el bit escondido es 0.

El menor exponente posible es, sesgado, 00001 y sin sesgar $1 - \text{sesgo} = 1 - 31 = -30$. Por lo tanto, el menor número desnormalizado es $(0).000001 \times 2^{-30}$ que en decimal es

$$2^{-6} \times 2^{-30} = 2^{-36} = 1.45519 \times 10^{-11}$$

Formato binario

Como el exponente se escribe 000000 y la mantisa es (0).000001, teniendo en cuenta el signo, se escribirá

$$0 \ 000000 \ 000001$$