DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
ESCUELA DE INGENIERÍA
PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

IIC2343 - Arquitectura de Computadores (I/2026)

Ayudantía 8

Ayudantes: Alberto Maturana (alberto.maturana@uc.cl), Mario Rojas (mario.denzel@estudiante.uc.cl), Ignacio Gajardo (gajardo.ignacio@uc.cl)

Pregunta 1: Representación de números reales

(a) Suponga que posee dos representaciones para números reales que puede utilizar para almacenar el número `1000011,1101b`. La primera corresponde a una representación de punto fijo con la siguiente estructura: 1 bit de signo, 7 bits de parte entera y 4 bits de parte fraccional. La segunda, en contraste, corresponde a una representación de punto flotante con la siguiente distribución: 1 bit de signo de significante, 8 bits de significante, 1 bit de signo de exponente y 3 bits de exponente. ¿Cuál de estas representaciones recomendaría para almacenar el número?

Solución: Primero, es útil ver ambas representaciones propuestas para el número.

- **Punto fijo =** $0_{\texttt{signo}}1000011_{\texttt{entero}}1101_{\texttt{fraccionario}}$
- **Punto flotante =** $0_{\texttt{ss}}10000111_{\texttt{s}}0_{\texttt{se}}110_{\texttt{e}}$

A partir de esto, es posible visualizar que en el caso de la representación de punto fijo no se pierden bits, mientras que en la representación de punto flotante se pierden los 3 bits menos significativos.
Por lo tanto, es recomendable utilizar la representación de punto fijo, dado que, al perder menos bits de información, es más preciso.

(b) **[I2 2025-2]** Suponga que imprime en la consola de su computador un valor que se encuentra almacenado en formato float del estándar IEEE754. Al imprimir, en la consola ve que aparece el valor "`NaN`". ¿Es posible, a partir de este resultado, obtener la secuencia de 32 bits exacta que representa este valor? Justifique.

Solución: No es posible. La representación del valor `NaN` del formato float tiene las siguientes características:

- Su exponente es igual a 11111111.

- **Al menos uno de sus bits de significante es distinto de cero.**

El último punto es relevante. Dado que cualquiera de los bits del significante es distinto de cero, al leer en consola el valor "`NaN`" no tenemos cómo saber cuál de los 23 bits es igual a 1.

Pregunta 2: Punto fijo y Punto flotante

(a) Transforme los siguentes números a su notación en Punto Fijo (Signo/Magnitud) de 16 bits (parte racional de 8 bits).

- $-25{,}65625$

Solución:

- **Signo [1 bit]** = `1b` (signo negativo)
- Parte entera = $25 = 2^4 + 2^3 + 2^0$ = `11001b` $\rightarrow$ (7 bits: `0011001b`)
- Parte fraccionaria = $0.65625 = 2^{-1}+2^{-3}+2^{-5}$ = `0,10101b` $\rightarrow$ (8 bits: `10101000b`)

Por lo tanto, la representación en punto fijo es:

$$-25{,}65625 = \texttt{1 0011001 10101000}$$

- $124{,}796875$

Solución:

- **Signo [1 bit]** = `0b` (signo positivo)
- Parte entera = $124 = 2^6 + 2^5 + 2^4 + 2^3 + 2^2$ = `1111100b`
- Parte fraccionaria = $0.796875 = 2^{-1} + 2^{-2} + 2^{-5} + 2^{-6}$ = `0,110011b` $\rightarrow$ (8 bits: `11001100b`)

Por lo tanto, la representación en punto fijo es:

$$124{,}796875 = \texttt{0 1111100 11001100}$$

(b) Transforme los siguientes números decimales al estándar `IEEE-754` de 32 bits.

- $89{,}75$

Solución:

- **Signo [1 bit]** = `0b` (signo positivo)
- Parte entera = $89 = 2^6 + 2^4 + 2^3 + 2^0$ = `1011001b`
- Parte fraccionaria = $0.75 = 2^{-1} + 2^{-2}$ = `0,11b`
- Notación científica = `1011001,11b` = `1,01100111b` $\cdot\ 2^6$
- **Significante [23 bits]** = `01100111000000000000000b`

- **Exponente [8 bits]** = `exp` $- 127 = 6 \Rightarrow$ `exp` $= 133 =$ `10000101b`

Por lo tanto, la representación en estándar `IEEE-754` es:

$$89,75 = \texttt{0 10000101 01100111000000000000000}$$

- 16,8125

Solución:

- **Signo [1 bit]** = `0b` (signo positivo)
- Parte entera $= 16 = 2^4 =$ `10000b`
- Parte fraccionaria $= 0.8125 = 2^{-1} + 2^{-2} + 2^{-4} =$ `0,1101b`
- Notación científica = `10000,1101b` = `1,00001101b` $\cdot\ 2^4$
- **Significante [23 bits]** = `00001101000000000000000b`
- **Exponente [8 bits]** = `exp` $- 127 = 4 \Rightarrow$ `exp` $= 131 =$ `10000011b`

Por lo tanto, la representación en estándar `IEEE-754` es:

$$16,8125 = \texttt{1 10000011 00001101000000000000000}$$

Pregunta 3: Suma y multiplicacion en `IEEE-754`

(a) Realice la suma entre 89,75 y 16,8125 y represente el resultado en `IEEE-754`

Solución: Recuerde que las representaciones de estos números en `IEEE-754` son:

- $89{,}75 = \texttt{0 10000101 01100111000000000000000} = \texttt{1,01100111b} \cdot 2^6$
- $16{,}8125 = \texttt{0 10000011 00001101000000000000000} = \texttt{1,00001101b} \cdot 2^4$

En el algoritmo de la suma en `IEEE-754` es requisito que los exponentes de cada número sean iguales. Para conseguir esto, se realizarán 2 *shift right* al número de menor exponente, de modo que la representación de -16,8125 ahora es:

$$16,8125 = \texttt{0,0100001101b} \cdot 2^6$$

Luego, el siguiente paso es sumar los significantes normalizados. En caso de que estos no tengan la misma cantidad de bits, se debe hacer *zero-padding*. Entonces, la suma es:

$$\begin{aligned} & 1,0110011100 \cdot 2^6 \\ + \; & 0,0100001101 \cdot 2^6 \\ = \; & 1,1010101001 \cdot 2^6 \end{aligned}$$

Sabemos que en el estándar `IEEE-754` el exponente se encuentra desfasado en 127. Entonces:

$$\texttt{exp} - 127 = 6 \Rightarrow \texttt{exp} = 133 = \texttt{10000101b}$$

Finalmente, viendo la resta entre los números es posible ver que el resultado será positivo. Por lo tanto, el resultado es:

$$0\ 10000101\ 10101010010000000000000$$

(b) Interprete los números `0x40A00000` y `0x4170000` y multiplíquelos.

Solución: Las representaciones de los números `0x40A00000` y `0x41700000` en el formato `IEEE-754` son:

- $\texttt{0x40A00000} = \texttt{0 10000001 01000000000000000000000}$
- $\texttt{0x41700000} = \texttt{0 10000010 11100000000000000000000}$

Para realizar la multiplicación, se hace uso del algoritmo visto en clases. En primer lugar, es posible ver que los exponentes no desfasados de los números son: 2 y 3 respectivamente,

por lo tanto, para obtener el nuevo exponente se utiliza la siguiente fórmula: $exp_a + exp_b +$ desfase:

$$2 + 3 + 127 = 132$$

Por lo tanto, se obtiene que el exponente es: 132 o `10000100b`. Luego, se deben multiplicar los significantes (es importante notar que para este cálculo se ignoran las comas):

```
   1,111 × 1,01
   1 111
  00 00x
 111 1xx
= 1001 011
```

Luego, la coma se desplaza tanto como indique la suma de las posiciones del multiplicando y el multiplicador, en este caso 5. Por lo tanto, el resultado de la multiplicación de los significantes es: `10,01011`. Entonces, el número resultante es: `10,01011` $\times 2^5$, el cual no está normalizado, por lo que se debe hacer un *shift left* para normalizarlo. Esto implica que el valor del exponente aumenta en 1, pasando de 132 a 133 o `10000101` en binario, mientras que el valor de la mantisa es: `1,001011`. Por lo tanto, el valor del número normalizado es: `1,001011` $\times 2^6$, por lo que su representación en el estándar `IEEE-754` es:

`0 10000101 00101100000000000000000`

El cual representa el valor 75, es decir, la multiplicación de 5×13

(c) Interprete los números `0x82D00000` y `0x7F800000` y multiplíquelos.

Solución: Las representaciones de los números `0x82D00000` y `0x7F800000` en el formato `IEEE-754` son:

- `0x82D00000` = `1 00000111 10100000000000000000000`
  - **Signo:** `1b` → negativo.
  - **Exponente:** 00000111 → $7 - 127 = -120$
  - **Significante:** `10100000000000000000000b` → `1,101b`
  - **Valor:** $-(\texttt{1,101} \cdot 10^{-120})\texttt{b}$

- `0x7F800000` = `0 11111111 00000000000000000000000`
  - **Signo**: `0b` → positivo.
  - **Exponente**: `10000000b` → 255 - 127 = 128.
  - **Significante**: `00000000000000000000000b` → `1,0b`.
  - **Valor**: $+\infty$

De lo anterior, se deduce que la operación a realizar es $+\infty \times -(\texttt{1,101} \cdot 10^{-120})\texttt{b} \times 2 = -\infty$, cuya representación en el estándar `IEEE-754` es `0xFF800000`.

Feedback ayudantía

Escanee el QR para entregar feedback sobre la ayudantía.

