

Arquitectura de Computadoras 2026-1

Tarea 1

February 6, 2026

Expediente	Nombre
219208106	Bórquez Guerrero Angel Fernando

- Realizar las siguientes conversiones

- Convertir el entero -72 de base 10 a binario con 8 bits.

- **Conversión a binario**

Cociente / 2	Cociente	Residuo
72/2	36	0
36/2	18	0
18/2	9	0
9/2	4	1
4/2	2	0
2/2	1	0
1/2	0	1

$$72_{10} = 01001000_2$$

- **Complemento a 2**

Número base	01001000
Intercambio	10110111
+1	10111000

- **Resultado**

$$R/: -72_{10} = 10111000_2$$

- Convertir el entero 11001110 de base 2 con 8 bits a base 10.

- **Desarrollamos la ecuación**

$$\begin{aligned}N &= (1 * 2^7) + (1 * 2^6) + (0 * 2^5) + (0 * 2^4) + (1 * 2^3) + (1 * 2^2) + (1 * 2^1) + (0 * 2^0) \\N &= (1 * 128) + (1 * 64) + (0 * 32) + (0 * 16) + (1 * 8) + (1 * 4) + (1 * 2) + (0 * 1) \\N &= 128 + 64 + 0 + 0 + 8 + 4 + 2 + 0 \\N &= 206\end{aligned}$$

- **Resultado**

$$R/: 11001110_2 = 206_{10}$$

c) Convertir el número 1010010.101 de base 2 a base 16.

- Se agregan ceros necesarios para bloques de 4 dígitos

$$1010010.101 = 01010010.1010$$

- Conversión de bloques de 4 dígitos

$$0101 = 5$$

$$0010 = 2$$

$$1010 = A$$

- Resultado

$$R/: 1010010.101_2 = 52.A_{16}$$

d) Convertir el número 48.D de base 16 a base 2.

- Conversión de dígitos a base 2

$$4 = 0100$$

$$8 = 1000$$

$$D = 1101$$

- Resultado

$$R/: 48.D_{16} = 1001000.1101_2$$

e) Mostrar cómo se representa el número de punto flotante -121.47 (base 10) en el standard IEEE-754 con precisión sencilla.

- Bit de signo

Dado que el número es negativo entonces el bit de signo es 1.

- Conversión a binario

- Parte entera:

Cociente / 2	Cociente	Residuo
121/2	60	1
60/2	30	0
30/2	15	0
15/2	7	1
7/2	3	1
3/2	1	1
1/2	0	1

$$121_{10} = 1111001_2$$

- Parte fraccionaria:

Cociente * 2	Entero	Fracción
0.47 * 2	0	94
0.94 * 2	1	88
0.88 * 2	1	76
0.76 * 2	1	52
0.52 * 2	1	04
0.04 * 2	0	08
0.08 * 2	0	16
0.16 * 2	0	32
0.32 * 2	0	64
0.64 * 2	1	28
0.28 * 2	0	56
0.56 * 2	1	12
0.12 * 2	0	24
0.24 * 2	0	48
0.48 * 2	0	96
0.96 * 2	1	92
0.92 * 2	1	82

$$.47_{10} = .01111000010100011_2$$

- Normalización

$$1111001.01111000010100011 = 1.1110010111000010100011 * 2^6$$

- Exponente en exceso 127

- Exceso 127:

$$127 + 6 = 133$$

- Conversión a binario:

Cociente / 2	Cociente	Residuo
133/2	66	1
66/2	33	0
33/2	16	1
16/2	8	0
8/2	4	0
4/2	2	0
2/2	1	0
1/2	0	1

$$133_{10} = 10000101_2$$

- Resultado

$$\text{R/: } -121.47 = 11000010111100101111000010100011$$

2. Considerar dos implementaciones del mismo ISA. Las instrucciones se dividen en 4 clases: A, B, C y D.

- P1 tiene un reloj de 2 GHz y CPIs de 1, 2, 3 y 3.
- P2 tiene un reloj de 3 GHz y CPIs de 2, 2, 4 y 4.

Suponer que un programa hace siete mil millones de instrucciones con la siguiente mezcla:

A	B	C	D
10%	20%	50%	20%

Responer las siguientes preguntas:

a) ¿Cuál implementación es más rápida?

- **Calculamos el CPI de cada implementación**

$$CPI_{P1} = (1 * 0.1) + (2 * 0.2) + (3 * 0.5) + (3 * 0.2) = 0.1 + 0.4 + 1.5 + 0.6 = 2.6$$

$$CPI_{P2} = (2 * 0.1) + (2 * 0.2) + (4 * 0.5) + (4 * 0.2) = 0.2 + 0.4 + 2.0 + 0.8 = 3.4$$

- **Calculamos el tiempo de ejecución de cada implementación**

$$T_{P1} = \frac{7*10^9*2.6}{2*10^9} = 9.1s$$

$$T_{P2} = \frac{7*10^9*3.4}{3*10^9} \approx 7.93s$$

- **Respuesta**

R/: La implementación más rápida es P2.

b) ¿Cuál es el CPI del programa en cada implementación?

Utilizamos los valores obtenidos en el inciso anterior.

- **Respuesta**

R/: $CPI_{P1} = 2.6$ y $CPI_{P2} = 3.4$

c) ¿Cuántos ciclos hace el programa en cada implementación?

- **Obtenemos el número de instrucciones de cada clase**

$$A = 7 * 10^9 * 0.1 = 7 * 10^8$$

$$B = 7 * 10^9 * 0.2 = 14 * 10^8$$

$$C = 7 * 10^9 * 0.5 = 35 * 10^8$$

$$D = 7 * 10^9 * 0.2 = 14 * 10^8$$

- **Calculamos los CR de cada implementación**

$$CR_{P1} = (1 * 7 * 10^8) + (2 * 14 * 10^8) + (3 * 35 * 10^8) + (3 * 14 * 10^8)$$

$$CR_{P1} = 7 * 10^8 + 28 * 10^8 + 105 * 10^8 + 42 * 10^8 = 18.2 * 10^9$$

$$CR_{P2} = (2 * 7 * 10^8) + (2 * 14 * 10^8) + (4 * 35 * 10^8) + (4 * 14 * 10^8)$$

$$CR_{P2} = 14 * 10^8 + 28 * 10^8 + 140 * 10^8 + 56 * 10^8 = 23.8 * 10^9$$

- **Respuesta**

R/: $CR_{P1} = 18.2 * 10^9$ y $CR_{P2} = 23.8 * 10^9$

d) ¿Cuál es la velocidad pico de cada implementación?

- **Calculamos la velocidad pico de cada implementación utilizando su CPI mínimo**

$$CPI_{P1} \text{ mínimo} = 1$$

$$VP_{P1} = \frac{2GHz}{1} = 2GHz$$

$$CPI_{P2} \text{ mínimo} = 2$$

$$VP_{P2} = \frac{3GHz}{2} = 1.5GHz$$

- **Respuesta**

$$R/: VP_{P1} = 2GHz \text{ y } VP_{P2} = 1.5GHz$$

3. Suponer que, para un programa, el compilador A genera un número de instrucciones de $1.0 * 10^9$ y un tiempo de ejecución de 1.1s, mientras que el compilador B genera un número de instrucciones de $1.2 * 10^9$ y un tiempo de ejecución de 1.5s.

a) Encontrar el CPI promedio para cada programa, dado que el procesador tiene un ciclo de reloj de 1 ns.

- **Factorizamos el CPI**

$$T = \frac{Ni * CPI}{Vr}$$

$$T * Vr = Ni * CPI$$

$$CPI = \frac{T * Vr}{Ni}$$

- **CPI promedio de A**

$$CPI_A = \frac{1.1 * 10^9}{1 * 10^9} = 1.1$$

- **CPI promedio de B**

$$CPI_B = \frac{1.5 * 10^9}{1.2 * 10^9} = 1.25$$

- **Respuesta**

$$R/: CPI_A = 1.1 \text{ y } CPI_B = 1.25$$

b) Suponer que los programas compilados se ejecutan en dos procesadores diferentes. Si los tiempos de ejecución en ambos procesadores son iguales, ¿cuánto más rápido es el reloj del procesador que ejecuta el código del compilador A en comparación con el reloj del procesador que ejecuta el código del compilador B?

- c) Se desarrolla un nuevo compilador C que utiliza solo 6.0E8 instrucciones y tiene un CPI promedio 1.1. ¿Cuál es el speedup al usar este nuevo compilador en comparación con usar el compilador A o el compilador B en el procesador original?

- **Calculamos el tiempo del compilador C**

$$T = \frac{6 \times 10^8 \times 1.1}{1 \times 10^9} = 0.66s$$

- **Calculamos la mejora respecto a los compiladores A y B**

$$S_{A/C} = \frac{1.1}{0.66} \approx 1.667$$

$$S_{B/C} = \frac{1.5}{0.66} \approx 2.278$$

- **Respuesta**

R/: El compilador C es 66.7% más rápido respecto al compilador A y 127.8% respecto al compilador B.

4. Suponer que a una computadora se le cambia la tarjeta gráfica por otra y que un benchmark tardaba en correr 90 segundos usando la tarjeta original. Al correr el benchmark con la tarjeta nueva se detecta un speedup de 1.5. ¿Qué tan rápida es la nueva tarjeta gráfica si se sabe que el benchmark pasa el 40% del tiempo ejecutando instrucciones gráficas?

- a) Plantear claramente la ecuación a resolver.

- **Valores actuales**

$$t_{old} = 90s$$

$$S_{global} = 1.5$$

$$f_{mejora} = 0.4$$

- **Despejamos la ecuación para obtener el Speedup de la nueva tarjeta gráfica**

$$S_{global} = \frac{1}{1 - f_{mejora} + \frac{f_{mejora}}{S_{mejora}}}$$

$$S_{global} \left(1 - f_{mejora} + \frac{f_{mejora}}{S_{mejora}} \right) = 1$$

$$1 - f_{mejora} + \frac{f_{mejora}}{S_{mejora}} = \frac{1}{S_{global}}$$

$$\frac{f_{mejora}}{S_{mejora}} = \frac{1}{S_{global}} - 1 + f_{mejora}$$

$$\frac{f_{mejora}}{S_{mejora}} = \frac{1}{S_{global}} - \frac{S_{global}}{S_{global}} + \frac{f_{mejora} * S_{global}}{S_{global}}$$

$$\frac{f_{mejora}}{S_{mejora}} = \frac{1 - S_{global} + f_{mejora} * S_{global}}{S_{global}}$$

$$\frac{f_{mejora} * S_{global}}{S_{mejora}} = 1 - S_{global} + f_{mejora} * S_{global}$$

$$f_{mejora} * S_{global} = (1 - S_{global} + f_{mejora} * S_{global}) * S_{mejora}$$

$$S_{mejora} = \frac{f_{mejora} * S_{global}}{1 - S_{global} + f_{mejora} * S_{global}}$$

b) Resolver la ecuación correctamente.