

B) (10 pts.) Considere la siguiente secuencia de instrucciones de alto nivel:

$$a = b + c$$

$$b = a + c$$

$$d = a - b$$

Utilizando la técnica de **propagación por copia** para transformar esta secuencia hasta el punto en que ningún operando es un valor calculado. Anote las líneas en las que la transformación ha reducido el trabajo computacional y los casos en que aumentó.

Solución:

Procedimiento	Resumen
1. $a = b + c$	No hay transformación, por lo tanto, no hay cambio en el trabajo computacional.
2. $b = a + c$ $\rightarrow b = (b + c) + c$	De la línea 1 sabemos que $a = b + c$, entonces se sustituye. Note que la expresión resultante en esta línea 2 aumenta en agregándose una suma, por lo tanto, la transformación ha aumentado el trabajo computacional.
3. $d = a - b$ $\rightarrow d = (b + c) - ((b + c) + c)$ $\rightarrow d = -c$	A partir de las transformaciones aplicadas en las líneas 1 y 2 se realiza la sustitución. Note que en este caso en la línea 3 la transformación reduce la operación a una única expresión, por lo tanto, se reduce el trabajo computacional.

C) (20 pts.) Calcule el CPI efectivo para una implantación de una CPU RISC-V empleando la siguiente tabla y los programas indicados en la columna de equipo, por ejemplo, el equipo 4 utiliza un promedio de los datos de los programas *libquantum* y *mcf*.

Solución:

Calculando los promedios para el equipo 8, para las frecuencias de *astar* y *gcc*:

$$\begin{aligned} \text{Loads} &= \frac{26 + 17}{2} &= 21.5\% \\ \text{Stores} &= \frac{6 + 23}{2} &= 14.5\% \\ \text{Branches} &= \frac{18 + 20}{2} &= 19\% \\ \text{Jumps} &= \frac{2 + 4}{2} &= 3\% \\ \text{ALU Operations} &= \frac{45 + 34}{2} &= 39.5 \\ \text{Otras} &= \frac{3 + 2}{2} &= 2.5\% \\ \text{Total} &&100\% \end{aligned}$$

Entonces el CPI corresponde a:

$$CPI = \sum_{i=1}^n CPI_i \cdot \left(\frac{IC_i}{\text{Instruction count}} \right)$$

Se supone que el 60% de los Branch son tomados, entonces:

$$\rightarrow \text{taken branch} = 0.6 \cdot 19 = 11.4\%$$

$$\rightarrow \text{untaken branch} = 0.4 \cdot 19 = 7.6\%$$

$$CPI = (5 \cdot 21.5\%) + (3 \cdot 14.5\%) + (5 \cdot 11.4\%) + (3 \cdot 7.6\%) + (3 \cdot 3\%) + (1 \cdot 39.5\%) + (3 \cdot 2.5\%) =$$

$$CPI = 2.868$$

D) (10 pts.) Utilice el siguiente fragmento de código:

Loop: load	x1, 0(x2)	; load x1 from address x2 + 0
addi	x1, x1, 1	; x1++
sd	x1, 0, (x2)	; store x1 at address x2 + 0
addi	x2, x2, 4	; x2 = x2 + 4
sub	x4, x3, x2	; x4 = x3 - x2
bnez	x4, Loop	; branch to Loop if x4 != 0

Muestre la tabla de tiempos de la secuencia de instrucciones para un pipeline RISC de 5 etapas, similar a la figura C-8; no disponemos de hardware adicional para realizar “forwarding” ni “bypassing”, pero suponga que la lectura y escritura de los registros ocurre en el mismo ciclo y el valor es adelantado entre registros como se muestra en la figura C-5.

Solución:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
load	IF	ID	EX	MEM	WB																
addi		IF	Stall	Stall	ID	EX	MEM	WB													
sd					IF	Stall	Stall	ID	MEM	WB											
addi								IF	ID	EX	MEM	WB									
sub									IF	Stall	Stall	ID	EX	MEM	WB						
bnez												IF	Stall	Stall	ID	EX	MEM	WB			
load																IF	ID	EX	MEM	WB	