

(Subgrupos A) CUESTIONARIO PREVIO PARA ENTREGAR ANTES DE LA SESIÓN
SESIÓN PRÁCTICA 8: Jerarquía de Memoria IV – Cachés unificadas y separadas

Apellidos	Nombre	DNI	Fecha (dd/mm/aaaa)	Hora	Grupo

1. Dado el siguiente programa en ensamblador para MIPS con direcciones de 16 bits y sabiendo que las instrucciones están siempre almacenadas de forma consecutiva en memoria:

<pre> addi \$2, \$0, V; dirección inicial del vector addi \$3, \$2, 20 addi \$5, \$0, 0 loop: lw \$10, 0(\$2) add \$5, \$5, \$10 addi \$2, \$2, 4 slt \$4, \$2, \$3 bne \$4, \$0, loop </pre>	<pre> int i, acc=0; // En registros. int v[5] = { ... }; // int → 4 bytes. for (i = 0; i<5; i++) acc = acc + v[i]; </pre>
---	---

Realice los siguientes apartados teniendo en cuenta lo siguiente:

Dirección de la primera instrucción	0x0160
Valor de la constante V	0x10D8

Sea un computador con un procesador **secuencial** con cachés **separadas** de correspondencia directa CB-WA de 32 Bytes con líneas de 8 Bytes cada una.

- a) Represente la decodificación de la dirección que realiza el controlador de caché:

ETIQUETA	ÍNDICE	DESPLAZAMIENTO
_____ BITS	_____ BITS	_____ BITS

- b) Muestre la memoria principal una vez cargada las instrucciones y los datos en memoria. Detalle aquellos bloques que contienen instrucciones o elementos del vector. En su parte derecha muestre de forma simplificada la estructura de ambas cachés y **utilícela en el apartado siguiente para indicar la carga de bloques** que se realiza.

Memoria Principal	Memoria Caché de Instrucciones
	Memoria Caché de Datos

c) Realice la traza de accesos de las 2 primeras iteraciones y simultáneamente rellene la caché del apartado anterior según los bloques a los que se acceden:

	Instrucción o Elemento del Vector	Dirección de Memoria	Dirección de Bloque	Línea	Acierto o Tipo de Fallo
	addi \$2, \$0, v				
	addi \$3, \$2, 20				
	addi \$5, \$0, 0				
Iteración 1	lw \$10, 0(\$2)				
	v[0]				
	add \$5, \$5,\$10				
	addi \$2,\$2,4				
	slt \$4, \$2,\$3				
	bne \$4,\$0, loop				
Iteración 2	lw \$10, 0(\$2)				
	v[1]				
	add \$5, \$5,\$10				
	addi \$2,\$2,4				
	slt \$4, \$2,\$3				
	bne \$4,\$0, loop				

2. Calcule la frecuencia de fallos que se obtendría tras la ejecución completa del programa.