Práctica 2 – Eficiencia de la Memoria Caché

1. REPRESENTAR Y ANALIZAR LA TASA DE FALLOS OBTENIDA PARA DOS TIPOS DE FUNCIONES DE CORRESPONDENCIA ELEGIDAS DE ENTRE CORRESPONDENCIA DIRECTA, ASOCIATIVA POR CONJUNTOS (DEFINIR EL NÚMERO DE CONJUNTOS) Y TOTALMENTE ASOCIATIVA.

PARA ELLO, CADA GRUPO DE TRABAJO VA A TENER EN CUENTA LO SIGUIENTE:

- CONSIDERAR UN SISTEMA CON UN ÚNICO PROCESADOR.
- REALIZAR LAS SIMULACIONES CON DOS TAMAÑOS DIFERENTES DE MEMORIA CACHÉ.
- ELEGIR PARA TODAS LAS SIMULACIONES UNA MISMA POLÍTICA DE REEMPLAZO: ALEATORIA, LRU, FIFO O LFU, PARA EL CASO DE CORRESPONDENCIA ASOCIATIVA.
- REALIZAR LAS SIMULACIONES ELIGIENDO DOS TRAZAS DIFERENTES. COMENTAR LA UTILIDAD DE LAS TRAZAS ELEGIDAS.

Una vez hechas las simulaciones, recoger en una tabla los resultados obtenidos y representarlos en una gráfica. En esta gráfica se recogerá la tasa de fallos frente al tamaño de memoria caché para todas las funciones de correspondencia elegidas

Elegiremos en nuestro caso, comparar entre la correspondencia directa y la totalmente asociativa. Usaremos para ello, las trazas NASA7 y COMP, y el algoritmo LRU.

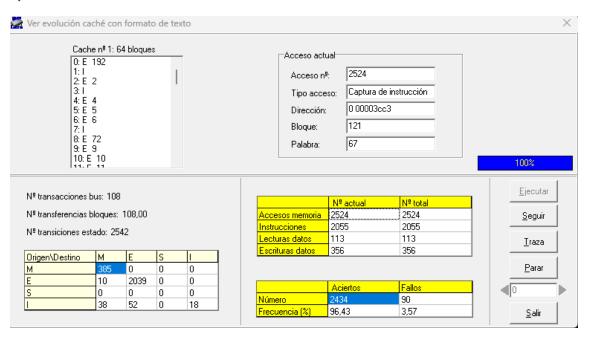
Tamaño de memoria caché de 32Kb:

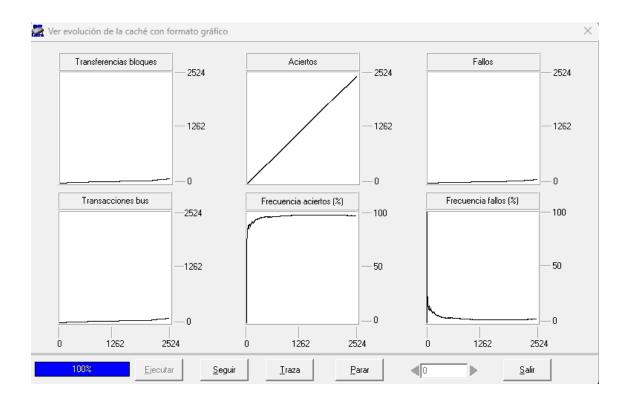
- Correspondencia directa:

Configuraremos la memoria caché de la siguiente forma:

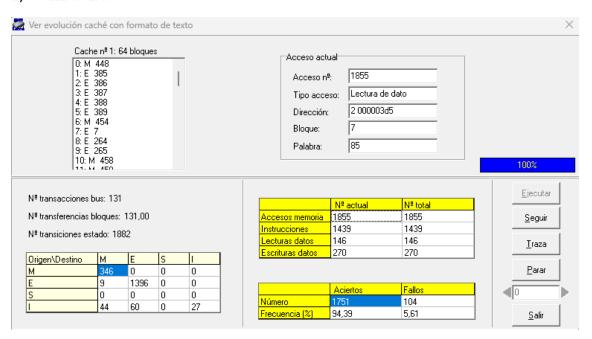
Configuración actual	
Bloques en caché:	64 💠
Tamaño de caché:	32 Kb
Función de correspondencia:	Directa 🔻
Nº de conjuntos:	NO 🛨
Algoritmo de reemplazamiento:	LRU 🔻
Niveles de caché:	1
Política de escritura:	Post-escritura
	Tamaño de caché: Función de correspondencia: Nº de conjuntos: Algoritmo de reemplazamiento: Niveles de caché:

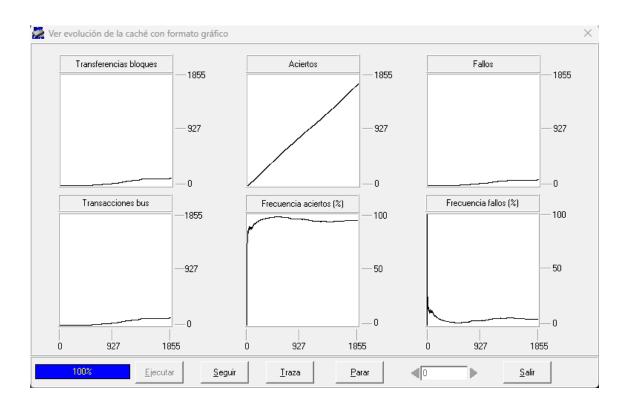
a) Traza COMP:





b) Traza NASA7:



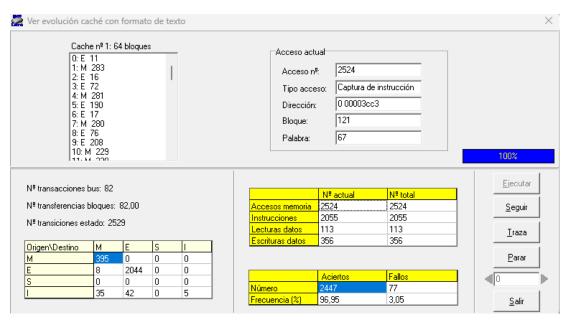


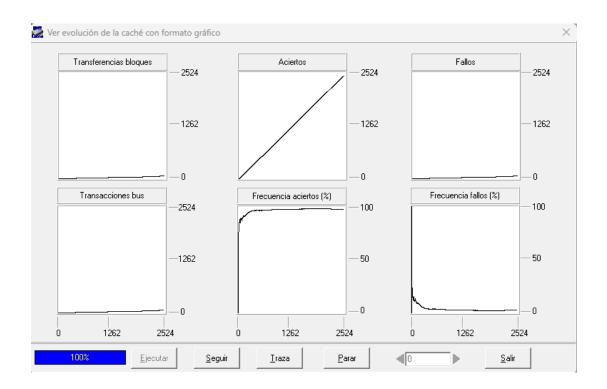
Correspondencia totalmente asociativa:

Configuraremos la memoria caché de la siguiente forma:

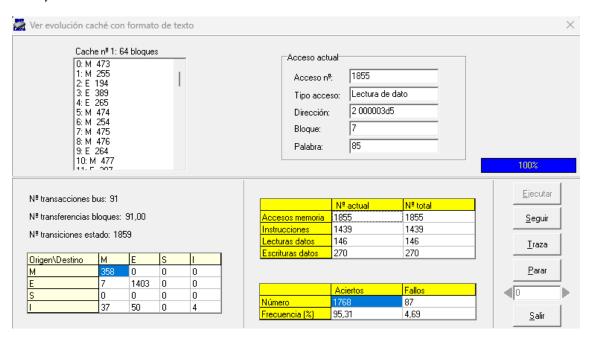


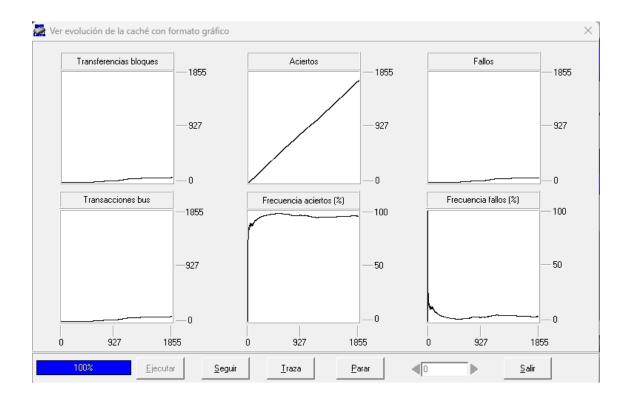
a) Traza COMP:



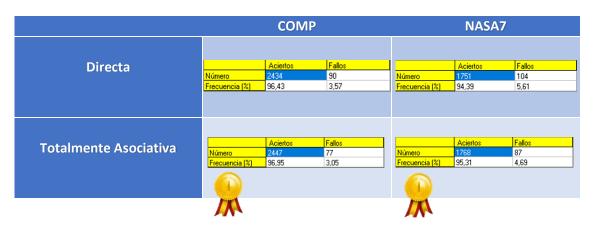


b) Traza NASA7:





- Resultado final:



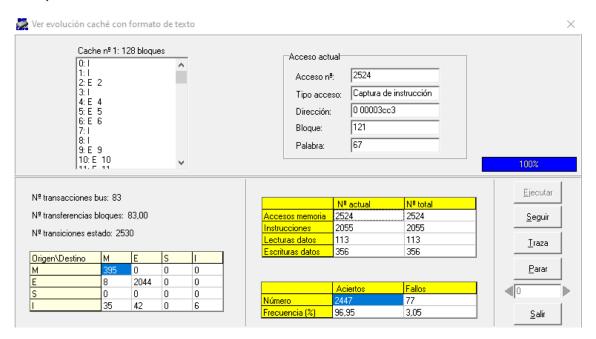
Tamaño de memoria caché de 64Kb:

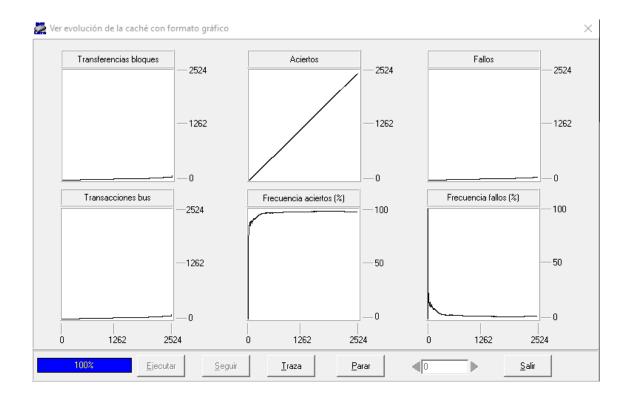
- Correspondencia directa:

Configuraremos la memoria caché de la siguiente forma:

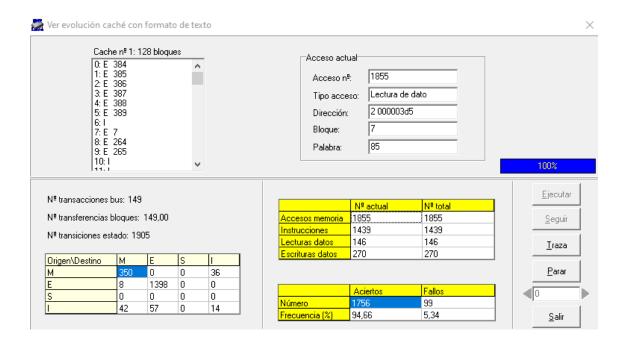


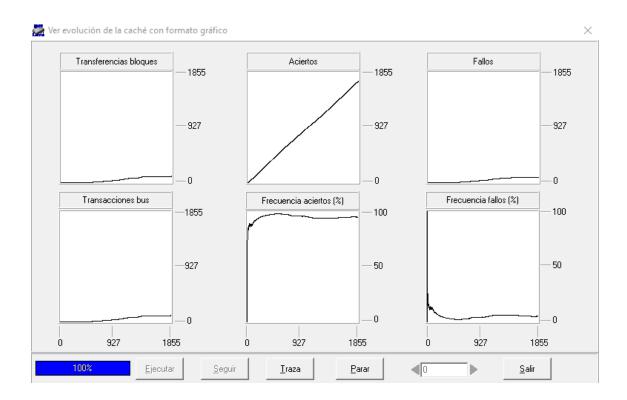
a) Traza COMP:





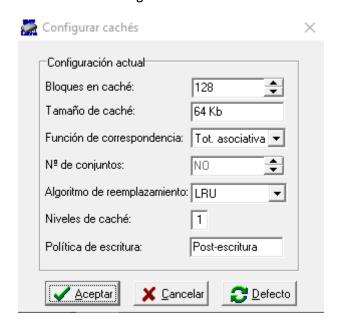
b) Traza NASA7:



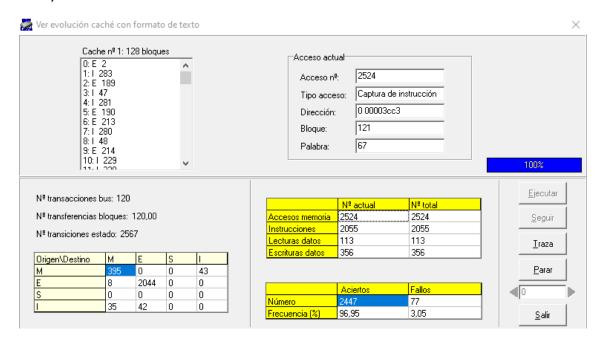


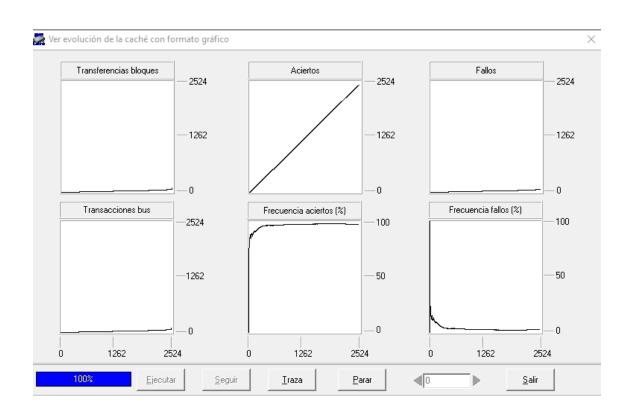
- Correspondencia totalmente asociativa:

Configuraremos la memoria caché de la siguiente forma:

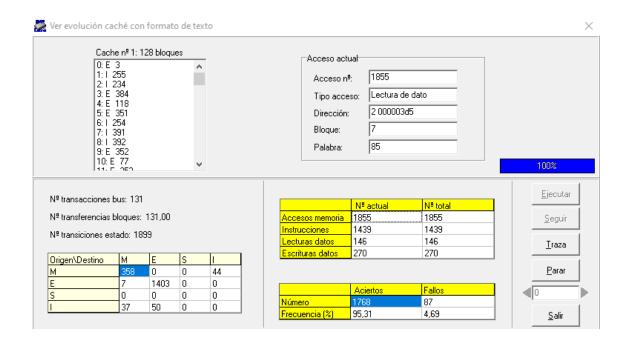


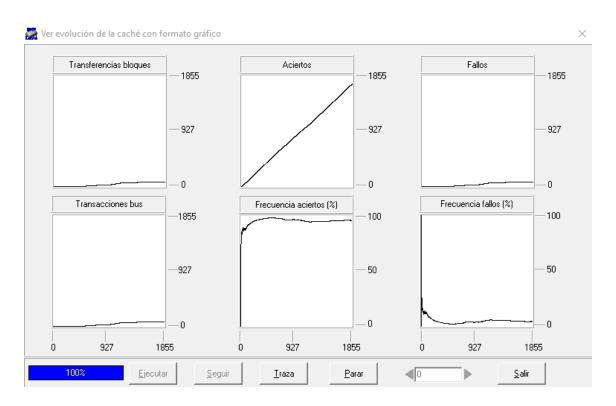
a) Traza COMP:



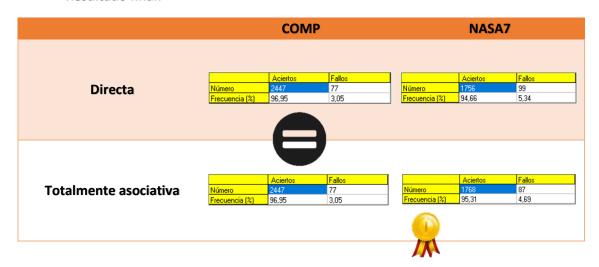


b) Traza NASA7:





- Resultado final:

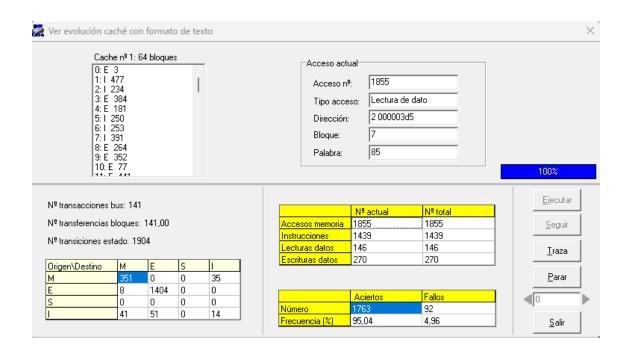


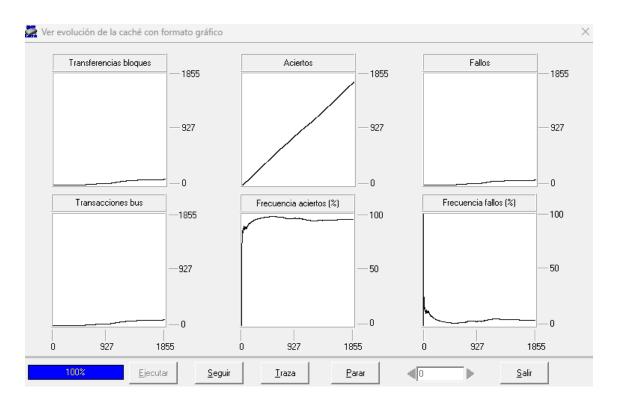
- 2. REALIZAR LAS SIMULACIONES PARA CADA ALGORITMO DE REEMPLAZO (ALEATORIO, LRU, FIFO Ó LFU) Y RESPONDER A LAS SIGUIENTES CUESTIONES:
 - ¿QUÉ ALGORITMO DE REEMPLAZO ES EL QUE OBTIENE MEJORES RESULTADOS?
 - ¿Y EL SEGUNDO MEJOR?
 - EMPLEANDO LA OPCIÓN DE VISIÓN DE LA EVOLUCIÓN DE LA CACHÉ EN MODO GRÁFICO, ¿SE MANTIENE CONSTANTE LA TASA DE FALLOS?, ¿POR QUÉ?

Para este apartado de la práctica, utilizaremos la configuración:

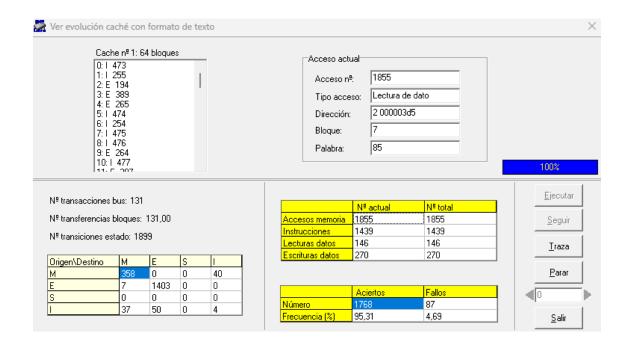


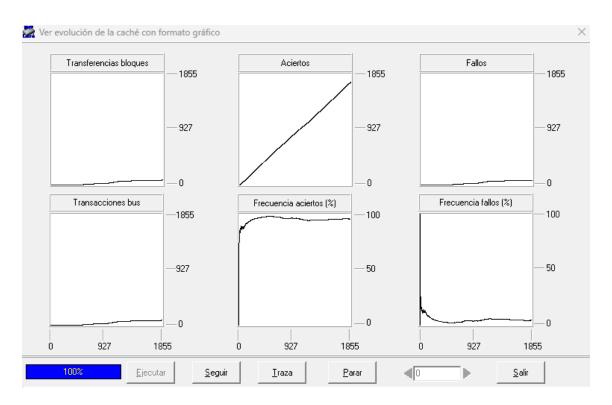
A) Aleatorio:



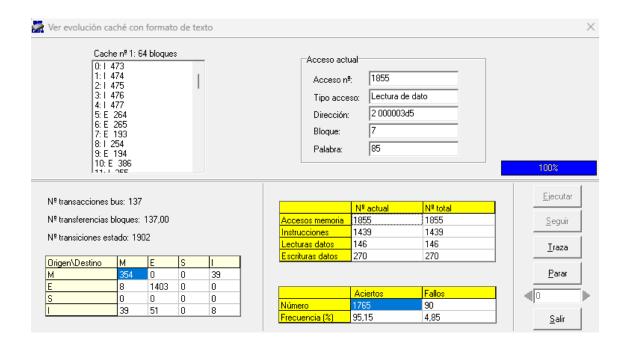


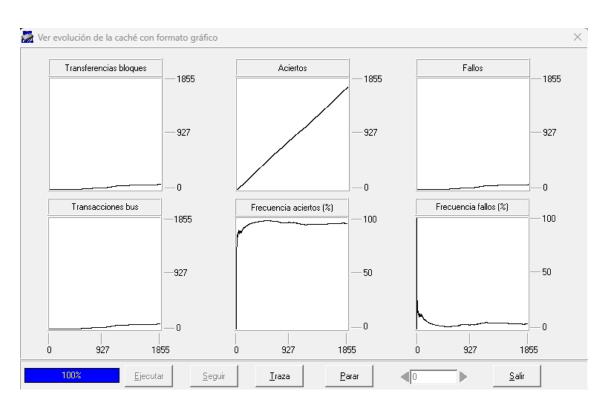
B) LRU:



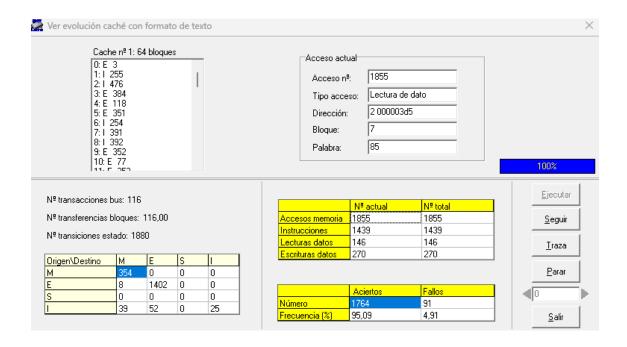


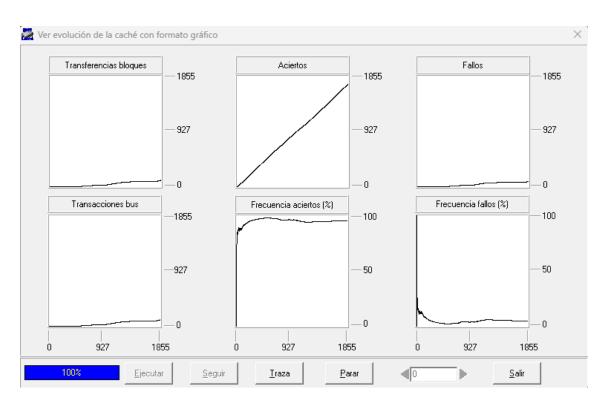
C) FIFO:





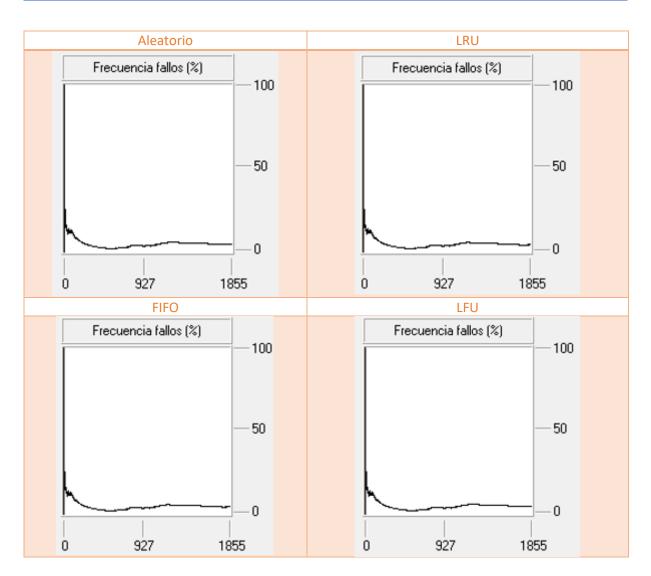
D) LFU:





- Comparación:

Aleatorio			LRU			Î	
	Aciertos	Fallos			Aciertos	Fallos	TX
Número	1763	92		Número	1768	87	
Frecuencia (%)	95,04	4,96		Frecuencia (%)	95,31	4,69	
FIFO			LFU				
	Aciertos	Fallos			Aciertos	Fallos	
Número	1765	90		Número	1764	91	
Frecuencia (%)	95,15	4,85		Frecuencia (%)	95,09	4,91	



Tras una alta tasa de fallos por cargar las primeras posiciones en memoria, esta se mantiene constante.

3. EN ESTA ÚLTIMA PARTE DE LA PRÁCTICA SE VAN A DISEÑAR TRES FICHEROS DE TRAZA DE MEMORIA DE FORMA QUE SE PUEDA DEMOSTRAR QUE DEPENDIENDO DEL PROGRAMA QUE SE ESTÉ EJECUTANDO, UNA ESTRUCTURA DE MEMORIA CACHÉ PUEDE DAR UN BUEN RENDIMIENTO CON UNA CONFIGURACIÓN DETERMINADA Y UN MAL RENDIMIENTO CON OTRA; AUNQUE DE MANERA GENERAL Y COMO MEDIA SE ESPERASE JUSTO LO CONTRARIO.

PARA ELLO SE VA A ESTABLECER LO SIGUIENTE:

- NÚMERO DE PROCESADORES: 1
- PROTOCOLO DE COHERENCIA: MESI
- ARBITRACIÓN DEL BUS: ALEATORIA.

Y CADA GRUPO DE TRABAJO VA A DEFINIR LO SIGUIENTE:

- TAMAÑO DE PALABRAS.
- TAMAÑO DE BLOQUE.
- TAMAÑO DE MEMORIA.
- TAMAÑO DE CACHE.

DISEÑAR TRES FICHEROS DE TRAZAS (FICHERO1.PRG, FICHERO2.PRG Y FICHERO3.PRG), CON 500 LÍNEAS CADA UNO DE ELLOS COMO MÍNIMO, QUE DEMUESTREN QUE:

- 1. FICHERO1.PRG → DA MEJOR RENDIMIENTO PARA UNA FUNCIÓN DE CORRESPONDENCIA DIRECTA QUE PARA UNA FUNCIÓN DE CORRESPONDENCIA TOTALMENTE ASOCIATIVA CON EL ALGORITMO DE REEMPLAZO ELEGIDO.
- 2. FICHERO2.PRG → DA MEJOR RENDIMIENTO PARA UN ALGORITMO DE REEMPLAZO QUE PARA OTRO (ELEGIR LOS DOS ALGORITMOS DE REEMPLAZO), CON UNA FUNCIÓN DE CORRESPONDENCIA ASOCIATIVA POR CONJUNTOS.
- 3. FICHERO3.PRG → OBTIENE EL MEJOR RENDIMIENTO JUSTO AL CONTRARIO QUE EL DEL FICHERO2.PRG.

MOSTRAR Y EXPLICAR LOS PATRONES SEGUIDOS POR CADA UNO DE LOS FICHEROS DE TRAZAS DISEÑADOS PARA DEMOSTRAR LO QUE SE PIDE EN EL ENUNCIADO.

MOSTRAR Y EXPLICAR LOS RESULTADOS QUE SE OBTIENEN AL SOMETERLOS AL PROCESO DE SIMULACIÓN CON.

NOTA: LOS FICHEROS CON TRAZAS DE MEMORIA TENDRÁN EXTENSIÓN .PRG, Y VAN A SER FICHEROS DE TEXTO. CONSTAN DE UN CONJUNTO DE LÍNEAS, CADA UNA DE LAS CUALES CONTIENE DOS NÚMEROS, SEPARADOS POR UN ÚNICO ESPACIO EN BLANCO: ETIQUETA VALOR DONDE:

ETIQUETA ES UN NÚMERO DECIMAL QUE IDENTIFICA EL TIPO DE OPERACIÓN DE ACCESO A MEMORIA DEMANDADO POR LA **CPU** EN UN MOMENTO DADO:

- CAPTURAR UNA INSTRUCCIÓN (0),
- LEER UN DATO DE MEMORIA (2)
- ESCRIBIR UN DATO EN MEMORIA (3).

VALOR ES UN NÚMERO HEXADECIMAL, QUE INDICA LA DIRECCIÓN EFECTIVA DE LA PALABRA DE MEMORIA A LA QUE SE VA A ACCEDER. ESTA DIRECCIÓN SERÁ TRADUCIDA POR EL SIMULADOR PARA LOCALIZAR LA PALABRA DENTRO DE LA ESTRUCTURA DE BLOQUES DEL SISTEMA DE MEMORIA.



Fichero1.prg

Patrón: B4, B9, B10, B15, B0

Correspondencia DIRECTA

Instante 1

MC

B4*	B4	B4	B4	-B4 B0*
	B9*	B9	B9	B9
		B10*	B10	B10
			B15*	B15

5 fallos, 0 aciertos

Instante 2

MC

80 B4*	B4	B4	B4	84 B0*
В9	B9	B9	B9	B9
B10	B10	B10	B10	B10
B15	B15 B15		B15	B15

2 fallos, 3 aciertos

Correspondencia TOTALMENTE ASOCIATIVA (FIFO)

Instante 1

MC

B4*	B4	B4	B4	B4 B0*
	B9*	B9	B9	B9
		B10*	B10	B10
			B15*	B15

5 fallos, 0 aciertos

Instante 2

MC

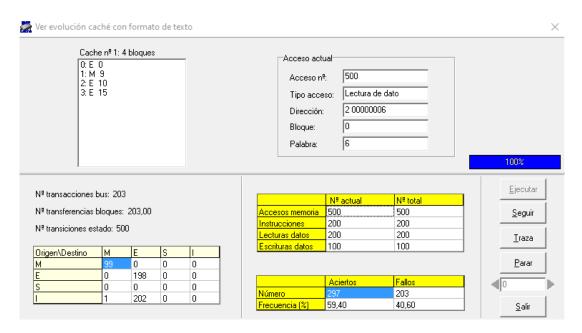
В0	B4	B4	B15*	B15
B4*	B4	B4	B4	B0*
B10	B9*	B9	B9	B9
B15	B15	B10*	B10	B10

5 fallos, 0 aciertos

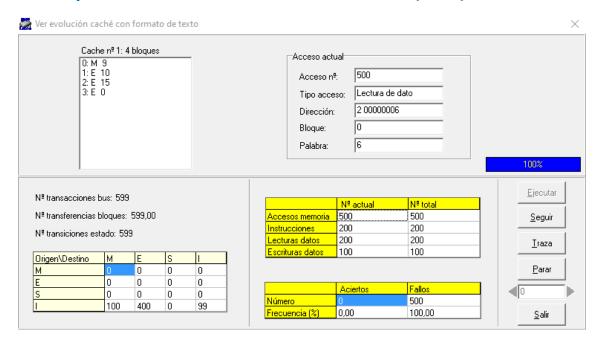
En la **correspondencia directa** solo nos daría 5 fallos en el primer instante. Para el resto de instantes solo se producirían 2 fallos y 3 aciertos constantemente. Por lo tanto, estimamos que habrá una tasa de **40**% de fallos. Sin embargo, para la **correspondencia totalmente asociativa** con el algoritmo **FIFO** vemos que se producen 5 fallos constantemente, es decir, una tasa de **100**% de fallos.

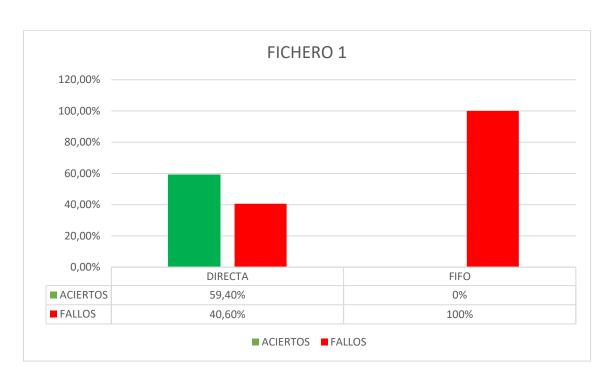
Vamos a comprobarlo con el simulador **SMPCACHE**:

Correspondencia DIRECTA



Correspondencia TOTALMENTE ASOCIATIVA (FIFO)



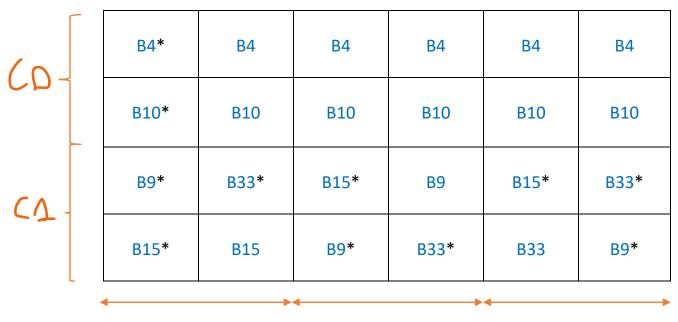


Fichero2.prg

Patrón: B4, B9, B10, B15, B4, B9, B10, B33

Correspondencia ASOCIATIVA POR CONJUNTOS (FIFO)

MC

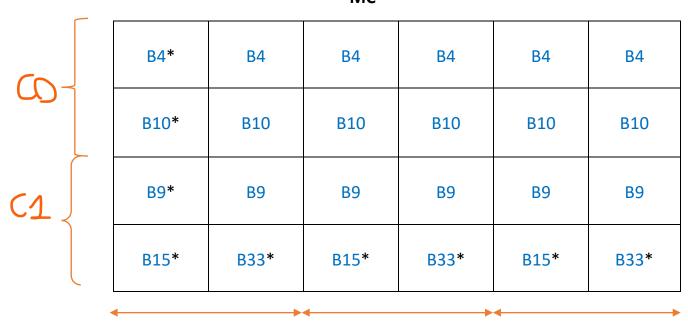


Instante 1 Instante 2 Instante 3

5 fallos y 3 aciertos en el 1er instante y 3 fallos y 5 aciertos en el resto

Correspondencia ASOCIATIVA POR CONJUNTOS (LRU)

MC

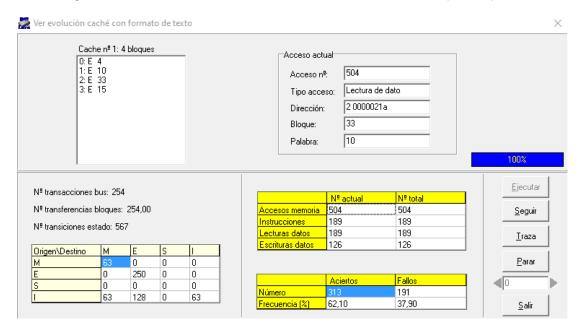


Instante 1 Instante 2 Instante 3

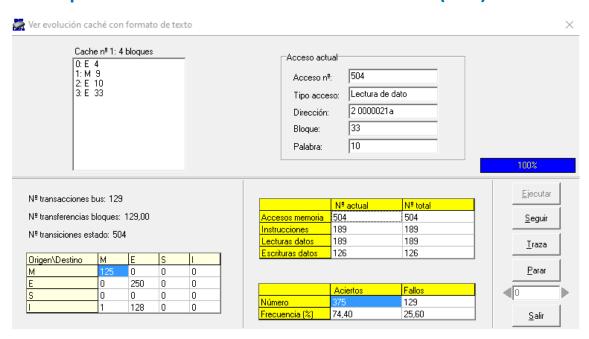
Como podemos observar, el algoritmo **LRU** es más eficiente con una tasa de fallos de aproximadamente **25%** (2 fallos y 6 aciertos), mientras que la FIFO cuenta con una tasa de fallos de **38%** (3 fallos y 5 aciertos)

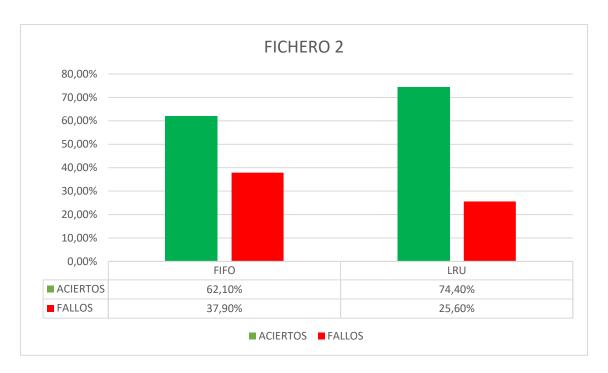
Vamos a comprobarlo con el simulador **SMPCACHE**:

Correspondencia ASOCIATIVA POR CONJUNTOS (FIFO)



Correspondencia ASOCIATIVA POR CONJUNTOS (LRU)



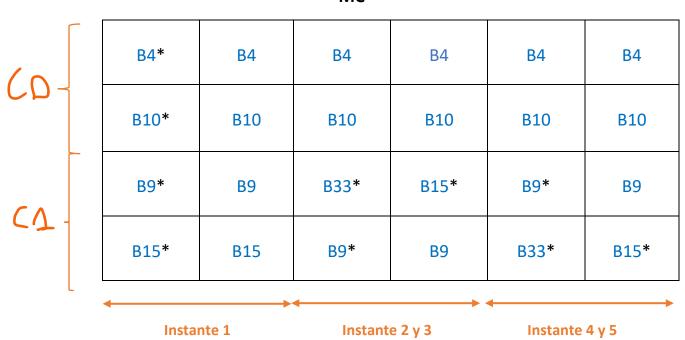


Fichero3.prg

Patrón: B4, B9, B10, B15, B15, B10, B9, B4, B33, B15, B10, B9, B4

Correspondencia ASOCIATIVA POR CONJUNTOS (FIFO)

MC



5 fallos y 4 aciertos en el 1er instante y 3 fallos y 7 aciertos en el resto cada 2 instantes

Correspondencia ASOCIATIVA POR CONJUNTOS (LRU)

Patrón: B4, B9, B10, B15, B15, B10, B9, B4 B33, B15, B10, B9, B4

MC

		B4*	B4	B4	B4	B4	B4
		B10*	B10	B10	B10	B10	B10
[1		B9*	B9	B15*	B9**	B15*	B33*
		B15*	B15	B9**	B15*	B9**	B9*
	•		•	•	-	•	

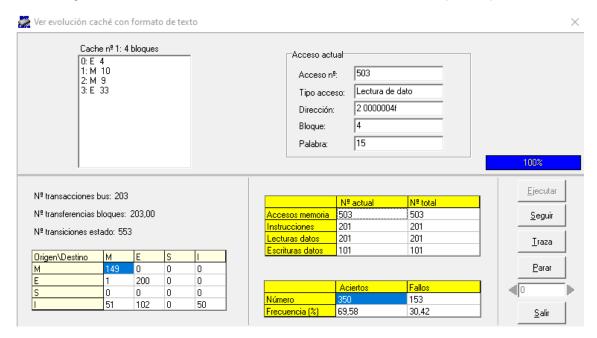
Instante 1 Instante 2 y 3 Instante 4 y 5

6 fallos y 2 aciertos en el 1er instante y para el resto 6 fallos y 4 aciertos cada 2 instantes

Como podemos observar, el algoritmo **FIFO** es más eficiente con una tasa de fallos de aproximadamente **30%** (3 fallos y 7 aciertos), mientras que la **LRU** cuenta con una tasa de fallos de **60%** (6 fallos y 4 aciertos)

Vamos a comprobarlo con el simulador **SMPCACHE**:

Correspondencia ASOCIATIVA POR CONJUNTOS (FIFO)



Correspondencia ASOCIATIVA POR CONJUNTOS (LRU)

