

Trabajo Práctico nº 2

Simulación de una Caché

Victor Belosevich Mugerli, *Padrón 97757*

Slack: Victor Belosevich Mugerli, victor.belosevich94@gmail.com

Valeria Rocha Bartaburu, *Padrón 90596*

Slack: Valeria Rocha, valeria.mrb@gmail.com

2do. Cuatrimestre de 2019

66.20 Organización de Computadoras – Práctica Jueves

Facultad de Ingeniería, Universidad de Buenos Aires

Resumen

Simulación de una memoria caché asociativa por conjuntos con política de reemplazo FIFO y política de escritura WB/WA.

1. Introducción

El objetivo principal del presente trabajo práctico es familiarizarse con el funcionamiento de la memoria caché implementando una simulación de una caché dada.

Se estudiará una caché asociativa por conjuntos, este tipo se divide en grupos de bloques (sets). Cada dirección de memoria se mapea en un único set de la caché, pero los datos pueden ubicarse en cualquiera de los bloques de ese set. Ésto último dependerá de la política de reemplazo utilizada. Si cada set posee 2 a la x bloques, la caché será 2 a la x ways associative.

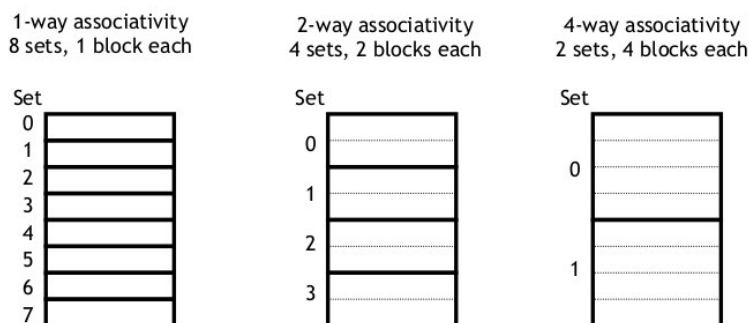


Figura 1: Ejemplo de configuraciones de caché.

La memoria a simular es una cache [1] asociativa por conjuntos de ocho vías, de 16KB de capacidad, bloques de 64 bytes, política de reemplazo FIFO y política de escritura WB/WA. Se asume que el espacio de direcciones es de 16 bits, y hay entonces una memoria principal a simular con un tamaño de 64KB. Estas memorias pueden ser implementadas como variables globales. Cada bloque de la memoria caché deberá contar con su metadata, incluyendo el tag, el bit V y la información necesaria para implementar la política de reemplazo FIFO.

2. Diseño e Implementación

Dado que se tiene un cache de ocho vias, de 16KB de capacidad, bloques de 64 bytes habrá 32 conjuntos. Los mismos se obtienen mediante el siguiente cálculo:

cantidad de conjuntos = capacidad total/cantidad de vías * tamaño de bloque
cantidad de conjuntos = $16\text{kB}/8 \times 64\text{B} = 32$

Se implementaron las siguientes primitivas:

```
void init()
unsigned int get_offset(unsigned int address)
unsigned int find_set(unsigned int address)
unsigned int select_oldest(unsigned int setnum)
void read_tocache(unsigned int blocknum, unsigned int way, unsigned int set)
void write_tomem(unsigned int blocknum, unsigned int way, unsigned int set)
unsigned char read_byte(unsigned int address)
void write_byte(unsigned int address, unsigned char value)
float get_miss_rate()
```

- La función `init()` debe inicializar la memoria principal simulada en 0, los bloques de la caché como inválidos y la tasa de misses a 0.
- La función `get_offset(unsigned int address)` debe devolver el offset del byte del bloque de memoria al que mapea la dirección `address`.
- La función `find_set(unsigned int address)` debe devolver el conjunto de caché al que mapea la dirección `address`.
- La función `select_oldest()` debe devolver la vía en la que está el bloque más “viejo” dentro de un conjunto, utilizando el campo correspondiente de los metadatos de los bloques del conjunto.
- La función `read_tocache(unsigned int blocknum, unsigned int way, unsigned int set)` debe leer el bloque `blocknum` de memoria y guardarlo en el conjunto y vía indicados en la memoria caché.
- La función `read_byte(unsigned int address)` debe buscar el valor del byte correspondiente a la posición `address` en la caché; si éste no se encuentra en la caché debe cargar ese bloque. El valor de retorno siempre debe ser el valor del byte almacenado en la dirección indicada.
- La función `write_byte(unsigned int address, unsigned char value)` debe escribir el valor `value` en la posición `address` de cache. Debe escribir en memoria cuando hay un reemplazo en la caché.

- La función `get miss rate()` debe devolver el porcentaje de misses desde que se inicializó la caché.
- `write tomem(unsigned int blocknum, unsigned int way, unsigned int set)`: implementa la funcionalidad de `write_back`

2.1. Desarrollo del Código Fuente

Tanto para simular la memoria principal como la cache, se utilizaron TDAs. En el caso de la memoria principal, el TDA está compuesto por un array de char. La memoria cache, por otro lado, está compuesta por un conjunto de sets. Los sets son otro TDA que consiste en un índice y un array de ways.

Los ways son la estructura que contiene a los bloques de data, el bit de validez, el bit de dirty, tag y 3 bits para la política de reemplazo (FIFO).

1. **Strutil**: esta librería nos ayuda con el manejo de strings, otorgando una implementación de la función split, la cual es usada en el parseo de los commands
2. **CommandParser**: se utiliza para el parseo de los comandos, también se encarga de validar la línea leída.
3. **FileReader**: es el encargado de leer el archivo de entrada línea por línea.
4. **AddressHelper**: esta librería se utiliza para la obtención del offset, el tag y el conjunto correspondiente a la dirección de memoria ingresada.

3. Proceso de Compilación

Se cuenta con un archivo Makefile que indica las reglas de compilación ejecutadas por el comando make.

```
CFLAGS= -mno-mips16 -mfp32 -mfp32 -gpubnames -mlong32 -mips1 -mabicalls
        -mlong-calls -mframe-header-opt -march=r2k

OBSJ= tp2

all: tp2 tp2_pf

prof: tp2_pf

tp2: main.c command.c cache.c set.c way.c fileReader.c commandParser.c
    commandExecutor.c addressHelper.c mainMemory.c strutil.c
    gcc -g $^ -o $@

tp2_pf: main.c command.c cache.c set.c way.c fileReader.c commandParser.c
    commandExecutor.c addressHelper.c mainMemory.c strutil.c
    gcc -no-pie -pg -g $^ -o $@

clean:
    rm tp2*
```

make tp2: Crea el ejecutable.

Para eliminar todos los archivos generados por el comando make, se puede ejecutar con la regla clean de la forma:

make clean

4. Casos de Prueba

Se realizaron ejecutaron los siguientes archivos de prueba:

1) Archivo: prueba1.mem

```
W 0, 255
W 2048, 048
W 4096, 096
W 8192, 192
W 16384, 084
W 32768, 068
W 12288, 088
W 20480, 080
R 0
R 2048
R 4096
R 8192
R 16384
R 32768
R 12288
R 20480
W 10240, 240
R 0
MR
```

Salida:

```
Se ley el comando: W 0, 255
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 255 en la
    direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 255 en la direccin: 0
Se ley el comando: W 2048, 048
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 48 en la
    direccin: 2048
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 2048 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 48 en la direccin: 2048
Se ley el comando: W 4096, 096
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 96 en la
    direccin: 4096
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 4096 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 96 en la direccin: 4096
Se ley el comando: W 8192, 192
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 192 en la
    direccin: 8192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
```

```
    a la direccin: 8192 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 192 en la direccin: 8192
Se ley el comando: W 16384, 084
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 84 en la
    direccin: 16384
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 16384 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 84 en la direccin: 16384
Se ley el comando: W 32768, 068
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 68 en la
    direccin: 32768
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 32768 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 68 en la direccin: 32768
Se ley el comando: W 12288, 088
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 88 en la
    direccin: 12288
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 12288 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 88 en la direccin: 12288
Se ley el comando: W 20480, 080
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 80 en la
    direccin: 20480
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 20480 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 80 en la direccin: 20480
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se ley el valor: 255 en la direccin: 0
Se ley el comando: R 2048
Se produjo un acceso a la memoria cache
Se ley el valor: 48 en la direccin: 2048
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se ley el valor: 96 en la direccin: 4096
Se ley el comando: R 8192
Se produjo un acceso a la memoria cache
Se ley el valor: 192 en la direccin: 8192
Se ley el comando: R 16384
Se produjo un acceso a la memoria cache
Se ley el valor: 84 en la direccin: 16384
Se ley el comando: R 32768
Se produjo un acceso a la memoria cache
Se ley el valor: 68 en la direccin: 32768
Se ley el comando: R 12288
Se produjo un acceso a la memoria cache
Se ley el valor: 88 en la direccin: 12288
Se ley el comando: R 20480
Se produjo un acceso a la memoria cache
Se ley el valor: 80 en la direccin: 20480
Se ley el comando: W 10240, 240
```

```
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 240 en la
    direccin: 10240
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 10240 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 240 en la direccin: 10240
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 0
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 2048
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 255 en la direccin: 0
Se ley el comando: MR
El miss rate es: 0.357
```

2) Archivo: prueba2.mem

```
R 0
R 31
W 64, 10
R 64
W 64, 20
R 64
MR
```

Salida:

```
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 0
Se ley el comando: R 31
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 31
Se ley el comando: W 64, 10
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 10 en la
    direccin: 64
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 64 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 10 en la direccin: 64
Se ley el comando: R 64
Se produjo un acceso a la memoria cache
Se ley el valor: 10 en la direccin: 64
```

```
Se ley el comando: W 64, 20
Se produjo un acceso a la memoria cache
Se coloc el valor: 20 en la direccin: 64
Se ley el comando: R 64
Se produjo un acceso a la memoria cache
Se ley el valor: 20 en la direccin: 64
Se ley el comando: MR
El miss rate es: 0.250
```

3) Archivo: prueba3.mem

```
W 0, 255
W 1, 2
W 2, 3
W 3, 4
W 4, 5
R 0
R 1
R 2
R 3
R 4
R 4096
R 8192
R 0
R 1
R 2
R 3
R 4
MR
```

Salida:

```
Se ley el comando: W 0, 255
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 255 en la
    direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 255 en la direccin: 0
Se ley el comando: W 1, 2
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 1
Se ley el comando: W 2, 3
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 2
Se ley el comando: W 3, 4
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 3
Se ley el comando: W 4, 5
Se produjo un acceso a la memoria cache
Se coloc el valor: 5 en la direccin: 4
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se ley el valor: 255 en la direccin: 0
Se ley el comando: R 1
```



```

Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 1
Se ley el comando: R 2
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 2
Se ley el comando: R 3
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 3
Se ley el comando: R 4
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 4
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4096
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 4096 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4096
Se ley el comando: R 8192
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 8192 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 8192
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se ley el valor: 255 en la direccin: 0
Se ley el comando: R 1
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 1
Se ley el comando: R 2
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 2
Se ley el comando: R 3
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 3
Se ley el comando: R 4
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 4
Se ley el comando: MR
El miss rate es: 0.150

```

3b) Archivo: prueba3b.mem

Se modificó la prueba para que se simule la misma ejecución dos veces. En este caso se espera que baje la tasa de miss rate ya que habrá mayor cantidad de accesos a memoria.

```

W 128, 1
W 129, 2
W 130, 3
W 131, 4
R 2176
R 4224
R 6272
R 8320
R 10368
R 12416
R 14464
R 16512

```

```
R 128
R 129
R 130
R 131
MR
W 128, 1
W 129, 2
W 130, 3
W 131, 4
R 2176
R 4224
R 6272
R 8320
R 10368
R 12416
R 14464
R 16512
R 128
R 129
R 130
R 131
MR
```

Salida:

```
Se ley el comando: W 128, 1
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 1 en la
    direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 128
Se ley el comando: W 129, 2
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 129
Se ley el comando: W 130, 3
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 130
Se ley el comando: W 131, 4
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 131
Se ley el comando: R 2176
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 2176
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 2176 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 2176
Se ley el comando: R 4224
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4224
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 4224 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4224
Se ley el comando: R 6272
Se produjo un acceso a la memoria cache
```

```
Se produjo un miss de lectura al intentar leer en la direccin: 6272
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 6272 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 6272
Se ley el comando: R 8320
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8320
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 8320 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 8320
Se ley el comando: R 10368
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 10368
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 10368 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 10368
Se ley el comando: R 12416
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 12416
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 12416 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 12416
Se ley el comando: R 14464
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 14464
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 14464 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 14464
Se ley el comando: R 16512
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 16512
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
correspondiente a la direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 16512 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 16512
Se ley el comando: R 128
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 128
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 128
Se ley el comando: R 129
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 129
Se ley el comando: R 130
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 130
Se ley el comando: R 131
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 131
```

```
Se ley el comando: MR
El miss rate es: 0.385
Empty line
Se ley el comando: W 128, 1
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 128
Se ley el comando: W 129, 2
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 129
Se ley el comando: W 130, 3
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 130
Se ley el comando: W 131, 4
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 131
Se ley el comando: R 2176
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 2176
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 2176 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 2176
Se ley el comando: R 4224
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4224
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 4224 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4224
Se ley el comando: R 6272
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 6272
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 6272 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 6272
Se ley el comando: R 8320
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8320
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 8320 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 8320
Se ley el comando: R 10368
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 10368
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 10368 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 10368
Se ley el comando: R 12416
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 12416
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
```

```
a la direccin: 12416 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 12416
Se ley el comando: R 14464
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 14464
Se ley el comando: R 16512
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 16512
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
correspondiente a la direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 16512 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 16512
Se ley el comando: R 128
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 128
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 128
Se ley el comando: R 129
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 129
Se ley el comando: R 130
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 130
Se ley el comando: R 131
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 131
Se ley el comando: MR
El miss rate es: 0.360
```

4) Archivo: prueba4.mem

```
W 128, 1
W 129, 2
W 130, 3
W 131, 4
R 2176
R 4224
R 6272
R 8320
R 10368
R 12416
R 14464
R 16512
R 128
R 129
R 130
R 131
MR
```

Salida:

```
Se ley el comando: W 128, 1
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 1 en la
  direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 128
Se ley el comando: W 129, 2
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 129
Se ley el comando: W 130, 3
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 130
Se ley el comando: W 131, 4
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 131
Se ley el comando: R 2176
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 2176
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 2176 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 2176
Se ley el comando: R 4224
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4224
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 4224 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4224
Se ley el comando: R 6272
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 6272
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 6272 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 6272
Se ley el comando: R 8320
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8320
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 8320 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 8320
Se ley el comando: R 10368
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 10368
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 10368 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 10368
Se ley el comando: R 12416
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 12416
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
  a la direccin: 12416 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 12416
Se ley el comando: R 14464
```

```

Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 14464
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 14464 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 14464
Se ley el comando: R 16512
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 16512
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 16512 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 16512
Se ley el comando: R 128
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 128
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 128
Se ley el comando: R 129
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 129
Se ley el comando: R 130
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 130
Se ley el comando: R 131
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 131
Se ley el comando: MR
El miss rate es: 0.385

```

5) Archivo: prueba5.mem

```

R 131072
R 4096
W 16384, 256
R 8192
R 4096
R 0
R 4096
MR

```

Salida:

```

Se ley el comando: R 131072
Invalid address
Invalid command
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4096
Se produjo un acceso a memoria principal para cargar el bloque correspondiente

```

```
    a la direccin: 4096  en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4096
Se ley el comando: W 16384, 256
Invalid command
Se ley el comando: R 8192
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 8192  en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 8192
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4096
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0  en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 0
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 4096
Se ley el comando: MR
El miss rate es: 0.375
```

6) Archivo: hitTest.mem

```
FLUSH
W 0, 0
W 1, 1
W 2, 2
W 3, 3
W 4, 4
W 5, 5
W 6, 6
W 7, 7
W 8, 8
W 9, 9
W 10, 10
W 11, 11
W 12, 12
W 13, 13
W 14, 14
W 15, 15
W 16, 16
W 17, 17
W 18, 18
W 19, 19
W 20, 20
W 21, 21
W 22, 22
W 23, 23
W 24, 24
W 25, 25
W 26, 26
W 27, 27
```


W 28, 28
W 29, 29
W 30, 30
W 31, 31
W 32, 32
W 33, 33
W 34, 34
W 35, 35
W 36, 36
W 37, 37
W 38, 38
W 39, 39
W 40, 40
W 41, 41
W 42, 42
W 43, 43
W 44, 44
W 45, 45
W 46, 46
W 47, 47
W 48, 48
W 49, 49
W 50, 50
W 51, 51
W 52, 52
W 53, 53
W 54, 54
W 55, 55
W 56, 56
W 57, 57
W 58, 58
W 59, 59
W 60, 60
W 61, 61
W 62, 62
W 63, 63
R 0
R 1
R 2
R 3
R 4
R 5
R 6
R 7
R 8
R 9
R 10
R 11
R 12
R 13
R 14
R 15
R 16
R 17
R 18
R 19
R 20
R 21
R 22
R 23

```
R 24
R 25
R 26
R 27
R 28
R 29
R 30
R 31
R 32
R 33
R 34
R 35
R 36
R 37
R 38
R 39
R 40
R 41
R 42
R 43
R 44
R 45
R 46
R 47
R 48
R 49
R 50
R 51
R 52
R 53
R 54
R 55
R 56
R 57
R 58
R 59
R 60
R 61
R 62
R 63
MR
```

Salida:

```
Se ley el comando: FLUSH
Se inicializo la cache
Se ley el comando: W 0, 0
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 0 en la
    direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 0 en la direccin: 0
Se ley el comando: W 1, 1
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 1
Se ley el comando: W 2, 2
```

```
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 2
Se ley el comando: W 3, 3
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 3
Se ley el comando: W 4, 4
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 4
Se ley el comando: W 5, 5
Se produjo un acceso a la memoria cache
Se coloc el valor: 5 en la direccin: 5
Se ley el comando: W 6, 6
Se produjo un acceso a la memoria cache
Se coloc el valor: 6 en la direccin: 6
Se ley el comando: W 7, 7
Se produjo un acceso a la memoria cache
Se coloc el valor: 7 en la direccin: 7
Se ley el comando: W 8, 8
Se produjo un acceso a la memoria cache
Se coloc el valor: 8 en la direccin: 8
Se ley el comando: W 9, 9
Se produjo un acceso a la memoria cache
Se coloc el valor: 9 en la direccin: 9
Se ley el comando: W 10, 10
Se produjo un acceso a la memoria cache
Se coloc el valor: 10 en la direccin: 10
Se ley el comando: W 11, 11
Se produjo un acceso a la memoria cache
Se coloc el valor: 11 en la direccin: 11
Se ley el comando: W 12, 12
Se produjo un acceso a la memoria cache
Se coloc el valor: 12 en la direccin: 12
Se ley el comando: W 13, 13
Se produjo un acceso a la memoria cache
Se coloc el valor: 13 en la direccin: 13
Se ley el comando: W 14, 14
Se produjo un acceso a la memoria cache
Se coloc el valor: 14 en la direccin: 14
Se ley el comando: W 15, 15
Se produjo un acceso a la memoria cache
Se coloc el valor: 15 en la direccin: 15
Se ley el comando: W 16, 16
Se produjo un acceso a la memoria cache
Se coloc el valor: 16 en la direccin: 16
Se ley el comando: W 17, 17
Se produjo un acceso a la memoria cache
Se coloc el valor: 17 en la direccin: 17
Se ley el comando: W 18, 18
Se produjo un acceso a la memoria cache
Se coloc el valor: 18 en la direccin: 18
Se ley el comando: W 19, 19
Se produjo un acceso a la memoria cache
Se coloc el valor: 19 en la direccin: 19
Se ley el comando: W 20, 20
Se produjo un acceso a la memoria cache
Se coloc el valor: 20 en la direccin: 20
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se ley el valor: 0 en la direccin: 0
Se ley el comando: R 1
```

```
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 1
Se ley el comando: R 2
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 2
Se ley el comando: R 3
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 3
Se ley el comando: R 4
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 4
Se ley el comando: R 5
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 5
Se ley el comando: R 6
Se produjo un acceso a la memoria cache
Se ley el valor: 6 en la direccin: 6
Se ley el comando: R 7
Se produjo un acceso a la memoria cache
Se ley el valor: 7 en la direccin: 7
Se ley el comando: R 8
Se produjo un acceso a la memoria cache
Se ley el valor: 8 en la direccin: 8
Se ley el comando: R 9
Se produjo un acceso a la memoria cache
Se ley el valor: 9 en la direccin: 9
Se ley el comando: R 10
Se produjo un acceso a la memoria cache
Se ley el valor: 10 en la direccin: 10
Se ley el comando: R 11
Se produjo un acceso a la memoria cache
Se ley el valor: 11 en la direccin: 11
Se ley el comando: R 12
Se produjo un acceso a la memoria cache
Se ley el valor: 12 en la direccin: 12
Se ley el comando: R 13
Se produjo un acceso a la memoria cache
Se ley el valor: 13 en la direccin: 13
Se ley el comando: R 14
Se produjo un acceso a la memoria cache
Se ley el valor: 14 en la direccin: 14
Se ley el comando: R 15
Se produjo un acceso a la memoria cache
Se ley el valor: 15 en la direccin: 15
Se ley el comando: R 16
Se produjo un acceso a la memoria cache
Se ley el valor: 16 en la direccin: 16
Se ley el comando: R 17
Se produjo un acceso a la memoria cache
Se ley el valor: 17 en la direccin: 17
Se ley el comando: R 18
Se produjo un acceso a la memoria cache
Se ley el valor: 18 en la direccin: 18
Se ley el comando: R 19
Se produjo un acceso a la memoria cache
Se ley el valor: 19 en la direccin: 19
Se ley el comando: R 20
Se produjo un acceso a la memoria cache
Se ley el valor: 20 en la direccin: 20
Se ley el comando: MR
```

El miss rate es: 0.023

7) Archivo: missTest

```
FLUSH
W 0, 0
W 64, 1
W 128, 2
W 192, 3
W 256, 4
W 320, 5
W 384, 6
W 448, 7
W 512, 8
W 576, 9
W 640, 10
W 704, 11
W 768, 12
W 832, 13
W 896, 14
W 960, 15
W 1024, 16
W 1088, 17
W 1152, 18
W 1216, 19
W 1280, 20
R 0
R 64
R 128
R 192
R 256
R 320
R 384
R 448
R 512
R 576
R 640
R 704
R 768
R 832
R 896
R 960
R 1024
R 1088
R 1152
R 1216
R 1280
MR
```

Salida:

```
Se ley el comando: FLUSH
Se inicializo la cache
Se ley el comando: W 0, 0
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 0 en la
    direccin: 0
```

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 0 en la direccin: 0
Se ley el comando: W 64, 1
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 1 en la direccin: 64
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 64 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 64
Se ley el comando: W 128, 2
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 2 en la direccin: 128
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 128 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 128
Se ley el comando: W 192, 3
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 3 en la direccin: 192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 192 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 192
Se ley el comando: W 256, 4
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 4 en la direccin: 256
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 256 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 256
Se ley el comando: W 320, 5
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 5 en la direccin: 320
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 320 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 5 en la direccin: 320
Se ley el comando: W 384, 6
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 6 en la direccin: 384
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 384 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 6 en la direccin: 384
Se ley el comando: W 448, 7
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 7 en la direccin: 448
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 448 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 7 en la direccin: 448

Se ley el comando: W 512, 8
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 8 en la direccin: 512
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 512 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 8 en la direccin: 512
Se ley el comando: W 576, 9
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 9 en la direccin: 576
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 576 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 9 en la direccin: 576
Se ley el comando: W 640, 10
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 10 en la direccin: 640
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 640 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 10 en la direccin: 640
Se ley el comando: W 704, 11
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 11 en la direccin: 704
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 704 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 11 en la direccin: 704
Se ley el comando: W 768, 12
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 12 en la direccin: 768
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 768 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 12 en la direccin: 768
Se ley el comando: W 832, 13
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 13 en la direccin: 832
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 832 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 13 en la direccin: 832
Se ley el comando: W 896, 14
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 14 en la direccin: 896
Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 896 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 14 en la direccin: 896
Se ley el comando: W 960, 15
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 15 en la direccin: 960

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 960 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 15 en la direccin: 960

Se ley el comando: W 1024, 16

Se produjo un acceso a la memoria cache

Se produjo un miss de escritura al intentar colocar el valor: 16 en la direccin: 1024

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 1024 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 16 en la direccin: 1024

Se ley el comando: W 1088, 17

Se produjo un acceso a la memoria cache

Se produjo un miss de escritura al intentar colocar el valor: 17 en la direccin: 1088

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 1088 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 17 en la direccin: 1088

Se ley el comando: W 1152, 18

Se produjo un acceso a la memoria cache

Se produjo un miss de escritura al intentar colocar el valor: 18 en la direccin: 1152

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 1152 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 18 en la direccin: 1152

Se ley el comando: W 1216, 19

Se produjo un acceso a la memoria cache

Se produjo un miss de escritura al intentar colocar el valor: 19 en la direccin: 1216

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 1216 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 19 en la direccin: 1216

Se ley el comando: W 1280, 20

Se produjo un acceso a la memoria cache

Se produjo un miss de escritura al intentar colocar el valor: 20 en la direccin: 1280

Se produjo un acceso a memoria principal para cargar el bloque correspondiente a la direccin: 1280 en memoria cache

Se produjo un acceso a la memoria cache

Se coloc el valor: 20 en la direccin: 1280

Se ley el comando: R 0

Se produjo un acceso a la memoria cache

Se ley el valor: 0 en la direccin: 0

Se ley el comando: R 64

Se produjo un acceso a la memoria cache

Se ley el valor: 1 en la direccin: 64

Se ley el comando: R 128

Se produjo un acceso a la memoria cache

Se ley el valor: 2 en la direccin: 128

Se ley el comando: R 192

Se produjo un acceso a la memoria cache

Se ley el valor: 3 en la direccin: 192

Se ley el comando: R 256

Se produjo un acceso a la memoria cache

Se ley el valor: 4 en la direccin: 256

Se ley el comando: R 320


```
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 320
Se ley el comando: R 384
Se produjo un acceso a la memoria cache
Se ley el valor: 6 en la direccin: 384
Se ley el comando: R 448
Se produjo un acceso a la memoria cache
Se ley el valor: 7 en la direccin: 448
Se ley el comando: R 512
Se produjo un acceso a la memoria cache
Se ley el valor: 8 en la direccin: 512
Se ley el comando: R 576
Se produjo un acceso a la memoria cache
Se ley el valor: 9 en la direccin: 576
Se ley el comando: R 640
Se produjo un acceso a la memoria cache
Se ley el valor: 10 en la direccin: 640
Se ley el comando: R 704
Se produjo un acceso a la memoria cache
Se ley el valor: 11 en la direccin: 704
Se ley el comando: R 768
Se produjo un acceso a la memoria cache
Se ley el valor: 12 en la direccin: 768
Se ley el comando: R 832
Se produjo un acceso a la memoria cache
Se ley el valor: 13 en la direccin: 832
Se ley el comando: R 896
Se produjo un acceso a la memoria cache
Se ley el valor: 14 en la direccin: 896
Se ley el comando: R 960
Se produjo un acceso a la memoria cache
Se ley el valor: 15 en la direccin: 960
Se ley el comando: R 1024
Se produjo un acceso a la memoria cache
Se ley el valor: 16 en la direccin: 1024
Se ley el comando: R 1088
Se produjo un acceso a la memoria cache
Se ley el valor: 17 en la direccin: 1088
Se ley el comando: R 1152
Se produjo un acceso a la memoria cache
Se ley el valor: 18 en la direccin: 1152
Se ley el comando: R 1216
Se produjo un acceso a la memoria cache
Se ley el valor: 19 en la direccin: 1216
Se ley el comando: R 1280
Se produjo un acceso a la memoria cache
Se ley el valor: 20 en la direccin: 1280
Se ley el comando: MR
El miss rate es: 0.333
```

8) Archivo: replacementTest

```
FLUSH
W 0, 1
R 0
W 2048, 2
R 2048
W 4096, 3
R 4096
```

```
W 6144, 4
R 6144
W 8192, 5
R 8192
W 10240, 6
R 10240
W 12288, 7
R 12288
W 14336, 8
R 14336
W 16384, 9
R 16384
R 0
R 2048
R 4096
R 6144
R 8192
R 10240
R 12288
R 14336
MR
```

Salida:

```
Se ley el comando: FLUSH
Se inicializo la cache
Se ley el comando: W 0, 1
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 1 en la
    direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 1 en la direccin: 0
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 0
Se ley el comando: W 2048, 2
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 2 en la
    direccin: 2048
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 2048 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 2 en la direccin: 2048
Se ley el comando: R 2048
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 2048
Se ley el comando: W 4096, 3
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 3 en la
    direccin: 4096
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 4096 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 3 en la direccin: 4096
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
```

```
Se ley el valor: 3 en la direccin: 4096
Se ley el comando: W 6144, 4
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 4 en la
    direccin: 6144
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 6144 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 4 en la direccin: 6144
Se ley el comando: R 6144
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 6144
Se ley el comando: W 8192, 5
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 5 en la
    direccin: 8192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 8192 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 5 en la direccin: 8192
Se ley el comando: R 8192
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 8192
Se ley el comando: W 10240, 6
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 6 en la
    direccin: 10240
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 10240 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 6 en la direccin: 10240
Se ley el comando: R 10240
Se produjo un acceso a la memoria cache
Se ley el valor: 6 en la direccin: 10240
Se ley el comando: W 12288, 7
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 7 en la
    direccin: 12288
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 12288 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 7 en la direccin: 12288
Se ley el comando: R 12288
Se produjo un acceso a la memoria cache
Se ley el valor: 7 en la direccin: 12288
Se ley el comando: W 14336, 8
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 8 en la
    direccin: 14336
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 14336 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 8 en la direccin: 14336
Se ley el comando: R 14336
Se produjo un acceso a la memoria cache
Se ley el valor: 8 en la direccin: 14336
Se ley el comando: W 16384, 9
Se produjo un acceso a la memoria cache
Se produjo un miss de escritura al intentar colocar el valor: 9 en la
    direccin: 16384
```

```
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 0
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 16384 en memoria cache
Se produjo un acceso a la memoria cache
Se coloc el valor: 9 en la direccin: 16384
Se ley el comando: R 16384
Se produjo un acceso a la memoria cache
Se ley el valor: 9 en la direccin: 16384
Se ley el comando: R 0
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 0
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 2048
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 0 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 1 en la direccin: 0
Se ley el comando: R 2048
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 2048
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 4096
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 2048 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 2 en la direccin: 2048
Se ley el comando: R 4096
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 4096
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 6144
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 4096 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 3 en la direccin: 4096
Se ley el comando: R 6144
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 6144
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 8192
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 6144 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 4 en la direccin: 6144
Se ley el comando: R 8192
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 8192
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 10240
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 8192 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 5 en la direccin: 8192
```

```

Se ley el comando: R 10240
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 10240
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 12288
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 10240 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 6 en la direccin: 10240
Se ley el comando: R 12288
Se produjo un acceso a la memoria cache
Se produjo un miss de lectura al intentar leer en la direccin: 12288
Todas las vias estn llenas, se procede a aplicar la politica de reemplazo
Se produjo un acceso a memoria principal para guardar el bloque
    correspondiente a la direccin: 16384
Se produjo un acceso a memoria principal para cargar el bloque correspondiente
    a la direccin: 12288 en memoria cache
Se produjo un acceso a la memoria cache
Se ley el valor: 7 en la direccin: 12288
Se ley el comando: R 14336
Se produjo un acceso a la memoria cache
Se ley el valor: 8 en la direccin: 14336
Se ley el comando: MR
El miss rate es: 0.381

```

4.1. Mediciones

Se realizaron mediciones con el comando `time` sobre las tres pruebas agregadas:

1. `missTest`
2. `hitTest`
3. `replacementTest`

Los resultados obtenidos fueron los siguientes:

Medición	missTest	hitTest	replacementTest
1	0,028	0,020	0,048
2	0,056	0,032	0,024
3	0,060	0,04	0,024
4	0,032	0,036	0,040
5	0,032	0,024	0,036
media	0,040	0,029	0,033

Cuadro 1: Tiempos de ejecución.

4.2. Análisis de resultados

En el primer caso se probó el tiempo que tarda tener todos miss de escritura. Este es el peor escenario posible ya que siempre se tiene que ir a buscar el bloque a la memoria principal.

En la segunda prueba se intentó reducir ese tiempo utilizando la cache implementada. Para aprovechar la localidad espacial se escribió en 20 bytes contiguos, todos del mismo bloque. Esto produjo que sólo haya una búsqueda contra la memoria principal, ya que la cache al inicio está vacía, generando un cold miss(fallo de arranque en frío).

En la última prueba se varió el tag pero no el index, haciendo que todas las escrituras fueran al mismo set. De esta forma pudimos probar que pasa cuando se llenan las vías y hay que aplicar la política de reemplazo (en este caso FiFo). Como en la prueba eran escrituras, el bit dirty se modificaba y era necesario guardar en memoria principal el bloque a reemplazar. Esto produce que se sume un miss de conflicto al ya mencionado de arranque en frío, derivando en un aumento de tiempo.

5. Conclusión

Pudimos probar mediante la simulación, que el ir a buscar los bloques siempre a memoria aumenta los tiempos. La cache acelera los tiempos de ejecución ya que escribir todo un bloque se ejecuta más rápido que colocar los datos en distintos bloques.

El uso de vías en una cache asociativa por conjuntos tiene sus ventajas y desventajas. La ventaja es que dependiendo el número de vías, nos permite disminuir la cantidad de miss de conflictos aumentando el rendimiento de la cache. La desventaja es que le agrega complejidad a las búsquedas en la cache, ya que en un conjunto vamos a tener más de un bloque y la búsqueda del mismo es secuencial.

6. Repositorio

<https://github.com/Belovic94/OrgaCompu-TP2>

Referencias

- [1] Hennessy, John L. and Patterson, David A., Computer Architecture: A Quantitative Approach, Third Edition, 2002.
- [2] B. Kernighan, D.Ritchie, *The C Programming Language*. Person Prentice Hall
- [3] Interaction Policies with Main Memory - <http://web.cs.iastate.edu/~prabhu/Tutorial/CACHE/interac.html>

7. Código fuente

main.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "fileReader.h"
#include "commandParser.h"
#include "cache.h"
#include "mainMemory.h"
#include "commandExecutor.h"

int main(int argc, char *argv[]) {
    FILE *fp;
    fp = fopen(argv[1], "r");
    if (fp == NULL)
        exit(EXIT_FAILURE);
    filereader_t file;
    filereader_create(&file, fp);
    if (cache_create() != 0 || main_memory_create() != 0) {
        filereader_destroy(&file);
        exit(EXIT_SUCCESS);
    }
    char *line = NULL;
    while (filereader_next(&file, &line) != -1) {
        if(empty_line(line) == 1) {
            fprintf(stdout, "Empty line \n");
            continue;
        }
        if ( line[strlen(line) -1] == '\n') {
            line[strlen(line) -1] = '\0';
        }
        printf("Se ley el comando: %s \n", line);
        command_t command;
        if (command_create(&command, line) != 0) {
            fprintf(stdout, "Invalid command\n");
            continue;
        }
        command_executor_execute(&command);
        command_destroy(&command);
    }
    cache_destroy();
    main_memory_destroy();
    filereader_destroy(&file);
    exit(EXIT_SUCCESS);
}
```


mainMemory.h

```
#ifndef ORGACOMPU_TP2_MAINMEMORY_H
#define ORGACOMPU_TP2_MAINMEMORY_H

typedef struct {
    unsigned char* memoryTable;
}main_memory_t;

int main_memory_create();

void main_memory_destroy();

void main_memory_read_tocache(unsigned int blockNum, unsigned char *block);

void main_memory_write_toMem(const unsigned char* block, unsigned int address);

#endif //ORGACOMPU_TP2_MAINMEMORY_H
```

mainMemory.c

```
#include "mainMemory.h"
#include "stdlib.h"
#include "addressHelper.h"
#include "stdio.h"

main_memory_t mainMemory;

int main_memory_create() {
    unsigned char* aTable = calloc(65536, sizeof(unsigned char));
    if (aTable == NULL) {
        return -1;
    }
    mainMemory.memoryTable = aTable;
    return 0;
}

void main_memory_destroy() {
    free(mainMemory.memoryTable);
}

void main_memory_read_tocache(unsigned int blockNum, unsigned char *block) {
    printf("Se produjo un acceso a memoria principal para cargar el bloque correspondiente "
           "a la direccin: %d en memoria cache \n", blockNum * BLOCK_SIZE);
    for (int i = 0; i < BLOCK_SIZE; ++i) {
        block[i] = mainMemory.memoryTable[(blockNum * BLOCK_SIZE) + i];
    }
}

void main_memory_write_toMem(const unsigned char *block, unsigned int address)
{
    printf("Se produjo un acceso a memoria principal para guardar el bloque correspondiente "
           "a la direccin: %d \n", address);
    for (int i = 0; i < BLOCK_SIZE; ++i) {
        mainMemory.memoryTable[address + i] = block[i];
    }
}
```

cache.h

```
#ifndef ORGACOMPU_TP2_CACHE_H
#define ORGACOMPU_TP2_CACHE_H

#include "set.h"
#include "addressHelper.h"

typedef struct {
    int access_counter;
    int miss_counter;
    set_t set[BLOCKS_NUMBER];
} cache_t;

int cache_create();

void cache_destroy();

void cache_init();

float cache_get_miss_rate();

int cache_read_byte(unsigned int address, unsigned char* byte_to_read);

int cache_write_byte(unsigned int address, unsigned char value);

unsigned int cache_get_free_way(unsigned int set);

void cache_save_block(unsigned char* block, unsigned int way, unsigned int set);

#endif //ORGACOMPU_TP2_CACHE_H
```

cache.c

```
#include "cache.h"
#include "stdio.h"

cache_t cache;

int cache_create() {
    cache.access_counter = 0;
    cache.miss_counter = 0;
    for (int i = 0; i < BLOCKS_NUMBER; ++i) {
        set_create(&cache.set[i], i);
    }
    return 0;
}

void cache_destroy() {
    for (int i = 0; i < BLOCKS_NUMBER; ++i) {
        set_destroy(&cache.set[i]);
    }
}

int cache_write_byte(unsigned int address, unsigned char value) {
    cache.access_counter++;
    printf("Se produjo un acceso a la memoria cache \n");
    if (set_write_byte(&cache.set[find_set(address)], address, value) != 0) {
```

```
        printf("Se produjo un miss de escritura al intentar colocar el valor: "
               "%u en la direccin: %u \n", value, address);
        cache.miss_counter++;
        return -1;
    } else {
        printf("Se coloc el valor: %u en la direccin: %u \n", value, address);
    }
    return 0;
}

float cache_get_miss_rate() {
    if (cache.access_counter == 0) {
        return 0;
    }
    return (float) cache.miss_counter / cache.access_counter;
}

int cache_read_byte(unsigned int address, unsigned char* byte_to_read) {
    cache.access_counter++;
    printf("Se produjo un acceso a la memoria cache \n");
    if (set_read_byte(&cache.set[find_set(address)], address, byte_to_read) ==
        -1) {
        printf("Se produjo un miss de lectura al intentar leer en la direccin:
        %u \n", address);
        cache.miss_counter++;
        return -1;
    }
    return 0;
}

unsigned int cache_get_free_way(unsigned int set) {
    return set_get_free_way(&cache.set[set]);
}

void cache_save_block(unsigned char *block, unsigned int way, unsigned int
    address) {
    set_save_block(&cache.set[find_set(address)], way, block, address);
}

void cache_init() {
    printf("Se inicializo la cache\n");
    cache.access_counter = 0;
    cache.miss_counter = 0;
    for (int i = 0; i < BLOCKS_NUMBER; ++i) {
        set_init(&cache.set[i]);
    }
}
```

set.h

```
#ifndef ORGACOMPU_TP2_QUEUE_H
#define ORGACOMPU_TP2_QUEUE_H

#include "way.h"
#include "addressHelper.h"

typedef struct {
    way_t ways[WAYS_NUMBER];
    unsigned int index: 5;
}set_t;

int set_create(set_t* self, int index);

void set_destroy(set_t* self);

int set_get_oldest(set_t* self);

int set_write_byte(set_t *self, unsigned int address, unsigned char value);

int set_read_byte(set_t* self, unsigned int address, unsigned char*
    byte_to_read);

unsigned int set_get_free_way(set_t* self);

void set_save_block(set_t* self, unsigned int way, unsigned char* block,
    unsigned int address);

void set_init(set_t* self);

#endif //ORGACOMPU_TP2_QUEUE_H
```

set.c

```
#include "set.h"
#include "stdio.h"

unsigned int _get_free_way_index(set_t* self);

int set_get_oldest(set_t *self) {
    int oldest = -1;
    unsigned int oldest_index = 0;
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        if (self->ways[i].old > oldest) {
            oldest_index = i;
            oldest = self->ways[i].old;
        }
    }
    return oldest_index;
}

void set_destroy(set_t *self) {
    for (int i = 0; i < WAYS_NUMBER ; ++i) {
        way_destroy(&self->ways[i]);
    }
}

int set_create(set_t *self, int index) {
```

```

    self->index = index;
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        way_create(&self->ways[i]);
    }
    return 0;
}

int set_write_byte(set_t *self, unsigned int address, unsigned char value) {
    unsigned int tag = get_tag(address);
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        if (self->ways[i].valid == 1 && self->ways[i].tag == tag) {
            way_write_byte(&self->ways[i], get_offset(address), value);
            return 0;
        }
    }
    return -1;
}

int set_read_byte(set_t *self, unsigned int address, unsigned char
*byte_to_read) {
    unsigned int tag = get_tag(address);
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        if (self->ways[i].valid == 1 && self->ways[i].tag == tag) {
            *byte_to_read = way_read_byte(&self->ways[i], get_offset(address));
            return 0;
        }
    }
    return -1;
}

unsigned int set_get_free_way(set_t* self) {
    unsigned int way_index = _get_free_way_index(self);
    if (way_index == -1) {
        printf("Todas las vias estn llenas, se procede a aplicar la politica
de reemplazo\n");
        way_index = set_get_oldest(self);
        way_write_back(&self->ways[way_index], self->index);
    }
    return way_index;
}

void set_save_block(set_t *self, unsigned int way, unsigned char *block,
    unsigned int address) {
    way_save_block(&self->ways[way], block, address);
}

void set_init(set_t *self) {
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        way_init(&self->ways[i]);
    }
}

unsigned int _get_free_way_index(set_t* self) {
    for (int i = 0; i < WAYS_NUMBER; ++i) {
        if (self->ways[i].valid == 0) {
            return i;
        }
    }
    if (self->ways[i].old < 7) {
        self->ways[i].old++;
    }
}

```

```

    }
}
return -1;
}

```

way.h

```

#ifndef ORGACOMPU_TP2_CACHE_BLOCK_H
#define ORGACOMPU_TP2_CACHE_BLOCK_H

typedef struct {
    unsigned int tag: 5;
    unsigned int valid : 1;
    unsigned int old: 3;
    unsigned int dirty: 1;
    unsigned char block[64];
}way_t;

void way_create(way_t* self);

void way_destroy(way_t* self);

unsigned char way_read_byte(way_t* self, unsigned int offset);

void way_write_byte(way_t* self, unsigned int offset, unsigned char value);

void way_save_block(way_t*, unsigned char* block, unsigned int address);

void way_write_back(way_t* self, unsigned int index);

void way_init(way_t* self);

#endif //ORGACOMPU_TP2_CACHE_BLOCK_H

```

way.c

```

#include "way.h"
#include "stdlib.h"
#include "addressHelper.h"
#include "mainMemory.h"

unsigned char way_read_byte(way_t *self, unsigned int offset) {
    return self->block[offset];
}

void way_create(way_t *self) {
    self->valid = 0;
    self->tag = 0;
    self->dirty = 0;
    self->old = 0;
}

void way_destroy(way_t *self) {

}

void way_save_block(way_t * self, unsigned char *block, unsigned int address) {

```

```

    for (int i = 0; i < BLOCK_SIZE; ++i) {
        self->block[i] = block[i];
    }
    self->valid = 1;
    self->tag = get_tag(address);
    self->old = 0;
    self->dirty = 0;
}

void way_write_back(way_t *self, unsigned int index) {
    if (self->dirty == 1) {
        unsigned int address = (self->tag << 11) | (index << 6);
        main_memory_write_toMem(self->block, address);
    }
}

void way_write_byte(way_t *self, unsigned int offset, unsigned char value) {
    self->dirty = 1;
    self->block[offset] = value;
}

void way_init(way_t *self) {
    for (int i = 0; i < BLOCK_SIZE; ++i) {
        self->block[i] = 0;
    }
    self->dirty = 0;
    self->old = 0;
    self->tag = 0;
    self->valid = 0;
}

```

addressHelper.h

```

#ifndef ORGACOMPU_TP2_ADDRESSHELPER_H
#define ORGACOMPU_TP2_ADDRESSHELPER_H

#define BLOCKS_NUMBER 32
#define BLOCK_SIZE 64
#define WAYS_NUMBER 8

unsigned int find_set(unsigned int address);
unsigned int get_offset(unsigned int address);
unsigned int get_tag(unsigned int address);
#endif //ORGACOMPU_TP2_ADDRESSHELPER_H

```

addressHelper.c

```
#include "addressHelper.h"

unsigned int find_set(unsigned int address) {
    unsigned int mba = address / BLOCK_SIZE;
    return mba % BLOCKS_NUMBER;
}

unsigned int get_offset(unsigned int address) {
    return address % BLOCK_SIZE;
}

unsigned int get_tag(unsigned int address) {
    return address >> 11;
}
```

command.h

```
#ifndef ORGACOMPU_TP2_COMMAND_H
#define ORGACOMPU_TP2_COMMAND_H
typedef struct {
    unsigned char operation;
    unsigned int address;
    unsigned char value;
}command_t;

int command_create(command_t* self, char* line);

void command_destroy(command_t* self);

#endif //ORGACOMPU_TP2_COMMAND_H
```

command.c

```
#include "command.h"
#include "commandParser.h"

int command_create(command_t* self, char* line){
    return parse(line, self);
}

void command_destroy(command_t* self){
}
```

commandExecutor.h

```
#ifndef ORGACOMPU_TP2_COMMANDEXECUTOR_H
#define ORGACOMPU_TP2_COMMANDEXECUTOR_H

#include "cache.h"
#include "mainMemory.h"
#include "command.h"

void command_executor_execute(command_t *command);
```



```
#endif //ORGACOMPU_TP2_COMMANDEXECUTOR_H
```

commandExecutor.c

```
#include "commandExecutor.h"
#include "addressHelper.h"
#include "stdio.h"

void init();
float get_miss_rate();
unsigned char read_byte(unsigned int address);
void write_byte(unsigned int address, unsigned char value);
void read_tocache(unsigned int way, unsigned int address);

void command_executor_execute(command_t *command) {
    switch (command->operation) {
        case 'F':
            init();
            break;
        case 'M':
            printf("El miss rate es: %.3f \n", get_miss_rate());
            break;
        case 'R':
            printf("Se ley el valor: %u en la direccin: %u \n",
                read_byte(command->address), command->address);
            break;
        case 'W':
            write_byte(command->address, command->value);
            break;
    }
}

void init() {
    cache_init();
}

float get_miss_rate() {
    return cache_get_miss_rate();
}

unsigned char read_byte(unsigned int address) {
    unsigned char byte_to_read;
    if(cache_read_byte(address, &byte_to_read) != 0) {
        read_tocache(cache_get_free_way(find_set(address)), address);
        cache_read_byte(address, &byte_to_read);
    }
    return byte_to_read;
}

void write_byte(unsigned int address, unsigned char value) {
    if (cache_write_byte(address, value) != 0) {
        read_tocache(cache_get_free_way(find_set(address)), address);
        cache_write_byte(address, value);
    }
}

void read_tocache(unsigned int way, unsigned int address) {
    unsigned char block[BLOCK_SIZE];
```

```

    main_memory_read_tocache(address / BLOCK_SIZE, block);
    cache_save_block(block, way, address);
}

```

commandParser.h

```

#ifndef ORGACOMPU_TP2_COMMANDPARSER_H
#define ORGACOMPU_TP2_COMMANDPARSER_H

#include "command.h"

int parse(char *split_line, command_t *command);
int empty_line(char *str);

#endif //ORGACOMPU_TP2_COMMANDPARSER_H

```

commandParser.c

```

#include <stdlib.h>
#include <stdio.h>
#include "commandParser.h"
#include "string.h"
#include "strutil.h"

#define INVALID_ADDRESS 65536

int _validate_value(int value) {
    if (value < 256 && value >= 0) {
        return 0;
    }
    return -1;
}

int _validate_address(int address) {
    if (address < INVALID_ADDRESS && address >= 0) {
        return 0;
    }
    fprintf(stdout, "Invalid address\n");
    return -1;
}

int _assign_address(char **split_line, int array_size, unsigned int* address) {
    if (array_size > 1) {
        char *address_line = split_line[1];
        if (split_line[0][0] == 'W') {
            strncpy(address_line, split_line[1], strlen(split_line[1]) - 1);
        }
        int address_aux = atoi(address_line);
        if (_validate_address(address_aux) == 0) {
            *address = (unsigned int) address_aux;
            return 0;
        }
    }
    return -1;
}

int _assign_value(char **split_line, int array_size, unsigned char* value){
    if(array_size > 2) {

```

```

        int value_aux = atoi(split_line[2]);
        if (_validate_value(value_aux) == 0) {
            *value = (unsigned char) value_aux;
            return 0;
        }
    }
    return -1;
}

int empty_line(char *str) {
    return strcmp(str, "\n") == 0 || strcmp(str, "\r\n") == 0;
}

int parse(char *line, command_t *command){
    int ret = 0;
    int array_size;
    char **split_line = split(line, ' ', &array_size);
    char operation = split_line[0][0];
    command->operation = operation;
    switch (operation) {
        case 'F':
            break;
        case 'M':
            break;
        case 'R':
            if (_assign_address(split_line, array_size, &command->address) !=
0) {
                ret = -1;
            }
            break;
        case 'W':
            if (_assign_address(split_line, array_size, &command->address) != 0
|| _assign_value(split_line, array_size, &command->value) != 0) {
                ret = -1;
            }
            break;

        default:
            ret = -1;
            break;
    }
    free_split(split_line);
    return ret;
}

```

fileReader.h

```

#ifndef ORGACOMPU_TP2_FILEREADER_H
#define ORGACOMPU_TP2_FILEREADER_H

#include <stdio.h>

typedef struct {
    FILE *fp;
}filereader_t;

void filereader_create(filereader_t* self, FILE* fp);

void filereader_destroy(filereader_t* self);

```

```
int filereader_next(filereader_t* self, char** line);

#endif //ORGACOMPU_TP2_FILEREADER_H
```

fileReader.c

```
#include <stdlib.h>
#include "fileReader.h"

void filereader_create(filereader_t *self, FILE *fp) {
    self->fp = fp;
}

void filereader_destroy(filereader_t *self) {
    fclose(self->fp);
}

int filereader_next(filereader_t *self, char** line) {
    size_t len = 0;
    return getline(line, &len, self->fp);
}
```

strutil.h

```
#ifndef ORGACOMPU_TP2_STRUTIL_H
#define ORGACOMPU_TP2_STRUTIL_H

char** split(const char* str, char sep, int * size);
void free_split(char **strv);

#endif //ORGACOMPU_TP2_STRUTIL_H
```

strutil.c

```
#include "strutil.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>

char* add_string(const char* str, size_t indice){
    char* str_aux = malloc(indice + 1);
    if (!str_aux) {
        return NULL;
    }
    strncpy(str_aux, str, indice);
    str_aux[indice] = '\0';
    return str_aux;
}

char** split(const char* str, char sep, int* size){
    char* vector[strlen(str)];
    char *pal_aux = strchr(str, sep);
    size_t i = 0;
    while (pal_aux != NULL) {
```

```
        vector[i] = add_string(str, strlen(str) - strlen(pal_aux));
        str = pal_aux + 1;
        pal_aux = strchr(str, sep);
        i++;
    }
    vector[i] = add_string(str, strlen(str));
    i++;
    vector[i] = NULL;
    //paso claves del vector local al vector hecho con malloc.
    char **strv = malloc((i + 1) * sizeof(char*));
    if (!strv) {
        return NULL;
    }

    for (int j = 0; j <= i ; j++){
        strv[j] = vector[j];
    }
    *size = i;
    return strv;
}

void free_split(char **strv){
    if (!strv) {
        return;
    }
    for (int i = 0; strv[i]; i++){
        free(strv[i]);
    }
    free(strv);
}
```