

MEMORIA PRÁCTICA 3 ARQO:

Como somos la pareja 10, el valor de P será 3 ya que es el valor de “10 mod 7”.

• Ejercicio 0:

Este ejercicio se divide en dos partes, la primera parte en la que se hará referencia a los datos relativos a la memoria caché de los equipos de los laboratorios, y la segunda en la que se analizará la arquitectura de nuestro equipo propio.

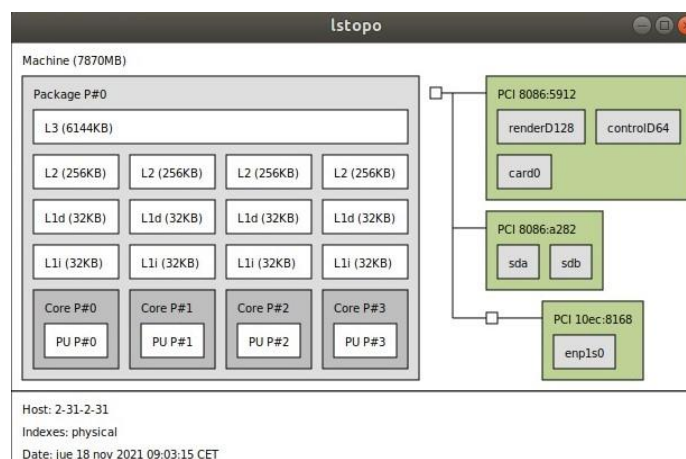
En lo referente a los datos de los equipos del laboratorio, tras ejecutar el comando “*getconf -a | grep -i cache*”, obtuvimos los siguientes resultados:

```
e420628@8B-14-66-47: ~
Archivo Editar Ver Buscar Terminal Ayuda
e420628@8B-14-66-47:~$ getconf -a | grep -i cache
LEVEL1_ICACHE_SIZE      32768
LEVEL1_ICACHE_ASSOC      8
LEVEL1_ICACHE_LINESIZE   64
LEVEL1_DCACHE_SIZE      32768
LEVEL1_DCACHE_ASSOC      8
LEVEL1_DCACHE_LINESIZE   64
LEVEL2_CACHE_SIZE        262144
LEVEL2_CACHE_ASSOC        4
LEVEL2_CACHE_LINESIZE    64
LEVEL3_CACHE_SIZE        16777216
LEVEL3_CACHE_ASSOC        16
LEVEL3_CACHE_LINESIZE    64
LEVEL4_CACHE_SIZE         0
LEVEL4_CACHE_ASSOC         0
LEVEL4_CACHE_LINESIZE     0
e420628@8B-14-66-47:~$
```

Esta estructura de la memoria caché se corresponde a la vista en teoría de los procesadores de Intel de séptima generación con 4 ‘cores’. Está compuesto por los niveles L1, L2 y L3:

- L1: Dividido en Datos e Instrucciones, con un tamaño de 32KB; funciona con un conjunto asociativo de 8 vías, con un offset de 64 Bytes por bloque.
- L2: Es un nivel de caché unificado, con un tamaño de 256KB; funciona con un conjunto asociativo de 4 vías, con un offset de 64 Bytes por bloque.
- L3: Es un nivel de caché unificado, con un tamaño de 16MB; funciona con un conjunto asociativo de 16 vías, con un offset de 64 Bytes por bloque.

Respecto a la segunda parte, analizaremos la arquitectura de uno de nuestros equipos particulares, a continuación, se mostrará la imagen resultante de ejecutar el siguiente comando “*lstopo*”.



Se puede observar que el cluster tiene asignado 8GB de memoria RAM, esto se puede apreciar en la esquina superior donde dice “Machine (8GB)”.

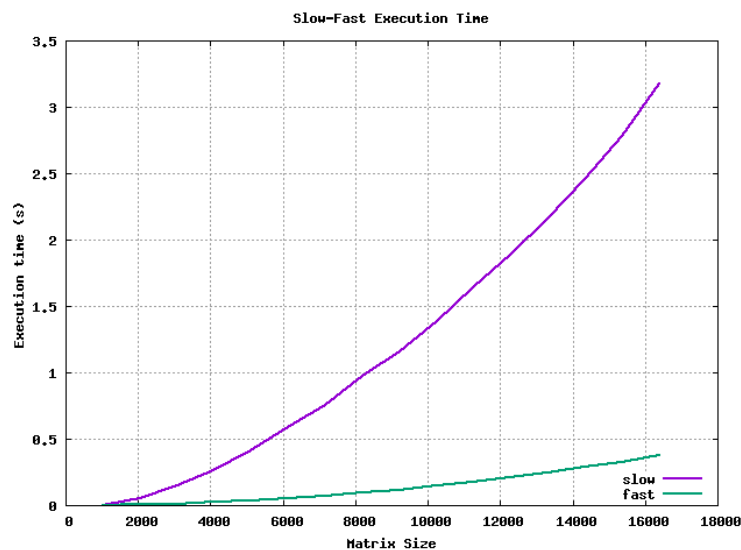
EL procesador como se puede ver en la imagen es de 4 'cores' y 3 niveles de memoria caché, con la estructura y el tamaño mencionado previamente para la otra imagen.

Por otro lado, se observan en verde las conexiones PCI de la placa, que se corresponden a los periféricos y demás conexiones.

• EJERCICIO 1:

- 2) El motivo por el que hay que realizar varias muestras para cada tamaño de cada matriz y programa es porque, para cada calculo el tiempo puede verse afectado por diferentes circunstancias y de ese modo en cada medida no se obtiene el mismo resultado de tiempo, es por ello por lo que realizar diferentes tomas para un cierto experimento hace que el error a la hora de tomar las medidas sea menor.

4)

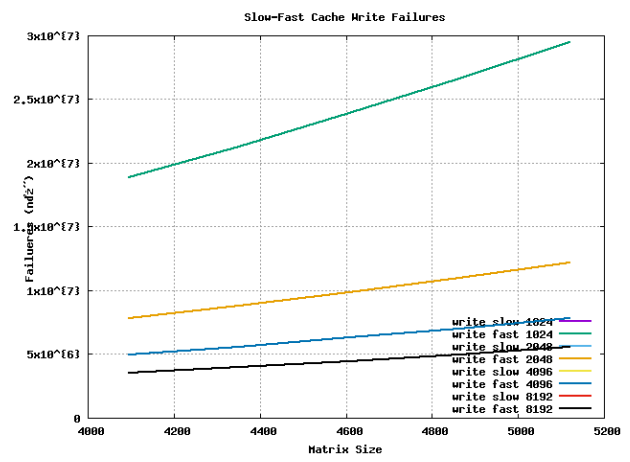
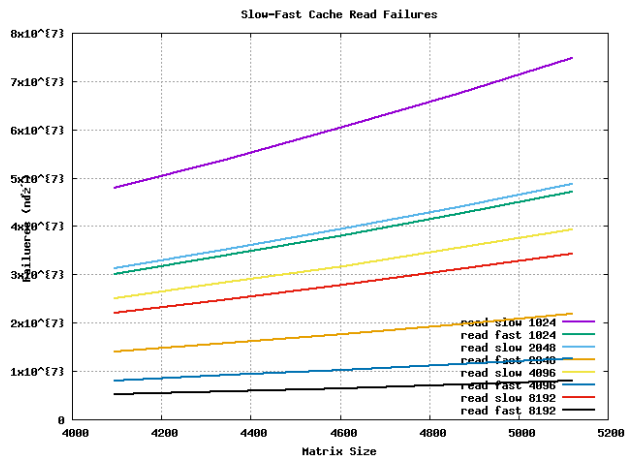


Se puede apreciar que según aumenta el tamaño de matriz, la ejecución lenta crece exponencial, esto se debe a que el archivo 'slow.c' accede a las posiciones de memoria de la matriz por columnas, mientras que el acceso por líneas es directo como se ve en el 'fast.c'

- 5) En nuestro caso, las matrices tienen dimensiones variables, lo que quiere decir que se declara un doble puntero, reservado memoria por tantas filas y recorreremos cada fila reservando memoria por cada columna, de modo que las filas (sus elementos) se almacenan de forma consecutiva pero no tiene por qué ocurrir entre diferentes filas.

• EJERCICIO 2:

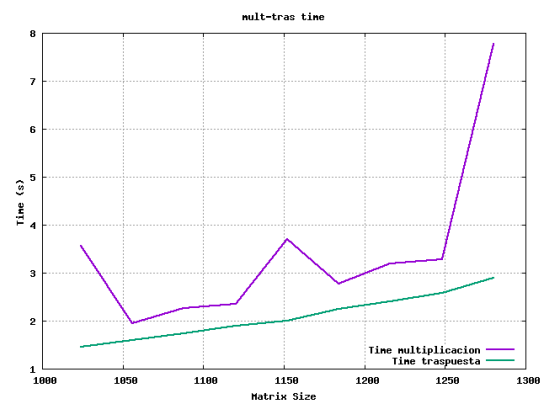
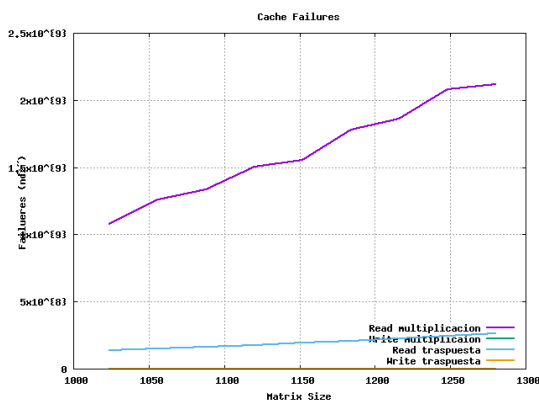
- 4) Tanto con el programa ./slow y ./fast a medida que el tamaño de la cache de nivel 1 aumenta, la tendencia de la gráfica tiende a disminuir, tener menos pendiente, esto quiere decir que el número de errores que se producen para cada ejecución según el tamaño de la cache de nivel 1 disminuye. Respecto a la comparación entre un tamaño de cache para la ejecución del programa ./slow y ./fast se aprecia como el número de errores que en el programa ./fast es menor.



La tendencia de las graficas se debe a que, al aumentar el tamaño del bloque, disminuye la tasa de fallos iniciales porque se produce una mejora de la localidad espacial (Palabras próximas en el espacio de memoria tienen una alta probabilidad de referenciar a palabras cercanas).

• EJERCICIO 3:

5)



La tasa de fallos va aumentando a medida que se aumenta el tamaño de la matriz debido a que la matriz resultante mientras quepa en caché, se mantendrá almacenada en ella, pero en el momento en el que ya no cabe, la tasa de fallos es mayor porque se encontrará accediendo a niveles superiores, al igual que el tiempo en realizar la operación, por el tiempo en acceder a esos niveles superiores de memoria.