Chapter III
Memory system

3

Memory hierarchy

Objectives

The main objective of the exercises of this unit is to show how the memory hierarchy provides a big capacity, high speed and low cost memory system, based on the principle of locality.

These exercises show a simplified version of the memory hierarchy. For instance, it is assumed that the access time when a memory hit occurs is negligible with regard to the access time produced by a miss. Thus, in a cache miss the required time to copy the piece of data from the cache to the CPU is negligible with regard to the time taken to copy the data block from the main memory to the cache.

When the disc level in the hierarchy is used, the block access time is considered constant independently of the block size, which is true with small blocks. In a disc, accessing the first byte of the block almost determines the whole access time, given the mechanical latencies involved to locate it.

Exercise 22.

The characteristics of a memory hierarchy are shown in table 3.1. The size of the cache memory block is 64 words. In each access to the cache, one word is transferred.

Answer the following questions:

❑ **22.1** Which is the cost of the memory system?

$$8 \times 15 + 8 \times 1024 \times 0.01 + 16 \times 1024 \times 0.0001 = 203.56 \ €$$

Level	t_{acc}	Size	Cost	Hit rate
Cache memory (c)	0.5 ns	8 MiB	15 €/MiB	0.99
Main memory (p)	3 ns	8 GiB	0.01 €/MiB	0.99999
Magnetic disc (d)	10 ms	16 GiB	0.0001 €/MiB	1

Table 3.1: Features of the memory hierarchy

❑ **22.2** Which is the average access time taking into account only the two closer memory levels to the processor, that is, assuming that the hit rate in the main memory is $100\,\%$? Answer in nanoseconds.

$$A_c \times t_c + (1 - A_c) \times t_p \times B = 0.99 \times 0.5 + 0.01 \times 3 \times 64 = 2.42\,\text{ns}$$

❑ **22.3** Which is the average access time taking all the memory levels into account? Answer in nanoseconds.

$$A_c \times t_c + (1 - A_c) \times (A_p \times t_p \times B + (1 - A_p) \times t_d) = 2.51\,\text{ns}$$

❑ **22.4** What would happen with the cache hit rate if the cache size were incremented? In the ideal case in which the hit rate were maximum, which would be the average access time to the memory system?

The cache hit rate would increase and the average memory access time would match the cache access time, that is, $0.5\,$ns.

❑ **22.5** What would happen with the cache hit rate if the cache size were reduced? Assuming that the cache hit rate is $75\,\%$ and the main memory hit rate is $100\,\%$, which is the average memory access time? In this scenario, would having a cache memory be of interest?

The hit rate would decrease. The average memory access time would be $48.38\,$ns with a hit rate of $75\,\%$. Thus, the access time would be increased notably. In this scenario it would not be beneficial having a cache memory, neither from the cost nor from the performance point of view.

❑ **22.6** What would happen with the number of disc accesses per time unit if the size of the main memory were increased? What would happen with the hierarchy access time in this scenario?

The number of disc accesses per time unit would decrease since the hit rate of the main memory would increase. Thus, the hierarchy access time would be reduced.

Exercise 23.

This exercise deduces the mathematical expression used to compute the average access time in a memory hierarchy, as well as the validity of some approximations.

Assume a memory hierarchy with the configuration shown in exercise 22. Taking this configuration into account, answer the following questions:

❑ **23.1** What are the three possible hit/miss scenarios in each level of the hierarchy? Example of answer: cache miss - main-memory miss - disk miss.

(a) Cache hit (main memory and disk are not accessed)
(b) Cache miss - main-memory hit (disk is not accessed)
(b) Cache miss - main-memory miss - disk hit

❑ **23.2** What is the average access time in a cache hit?

$t_{ac} = t_c = 0.5\,\mathrm{ns}$

❑ **23.3** What is the average access time when a cache miss occurs and the data is found in the main memory?

$t_{cm-ph} = t_p \times B_{cp} + t_c = 4 \times 32 + 0.5 = 128.5\,\mathrm{ns}$

❑ **23.4** Is it possible to do a reasonable approximation in the above computation? If so, compute the result again.

Cache access time can be ignored (it is really small in relation to the main memory access time). $t_{cm-ph} \approx t_p \times B_{cp} = 128\,\mathrm{ns}$

❑ **23.5** What is the average access time in a cache miss, main-memory miss, and disk hit?

$t_{cm-pm-dh} = t_d + t_p \times B_{cp} + t_c = 5 \times 10^6 + 4 \times 32 + 0.5 = 5000128.5\,\mathrm{ns}$

❑ **23.6** Is it possible to do a reasonable approximation in the above computation? If so, compute the result again.

Cache access time can be ignored. In addition, the time taken to copy a block from the main memory to the cache can also be ignored.
$t_{cm-pm-dh} \approx t_d = 5000000\,\mathrm{ns}$

❑ **23.7** All the memory accesses correspond to one of the sequences identified in the first question. Each sequence will have an occurrence probability depending on the cache and main memory hit rates. Assuming the cache and the main memory hit events are independent events, compute the probability for each sequence.

(a) $A_c = 0,992$
(b) $(1 - A_c) \times A_p = 0.992 \times 0.99999 \approx 0.992$
(c) $(1 - A_c) \times (1 - A_p) = 0.992 \times 0,00001 = 0,00000992$

❑ **23.8** Finally, taking the approximations identified for the average access times in cache, main memory and disk write down the expression used to compute the average access time in the system. As you can see, it is an average time since sometimes matches sequence (a), others sequence (b) and others sequence (c).

$$t_{cpd} = \begin{array}{l} A_c \times t_c + (1 - A_c) \times A_p \times t_p \times B_{cp} + (1 - A_c) \times (1 - A_p) \times t_d = \\ A_c \times t_c + (1 - A_c) \left[A_p \times t_p \times B_{cp} + (1 - A_p) \times t_d \right] \end{array}$$

Exercise 24.

A memory system design for a PC consists of three levels: cache memory, main memory and disc. The features of each memory level are the following:

- Cache. t_c: average access time to a 64-bit word is $0.5\,$ns.
- Main memory. t_p: average access time to a 64-bit word is $15\,$ns.
- Disc. t_d: average read time of any block between 1 byte and 10 Kbytes is $8\,$ms.

It is also known:

- Cache hit rate, A_c, is 98.5%.
- Main memory hit rate, A_p, is 99.995%.
- Cache block size is 16 words, 64 bits each.
- Writing a memory location involves writing simultaneously in cache and in main memory to keep up to date the stored data in both levels.

Taking all the above into account, answer the following questions.

❑ **24.1** Which is the average access time, t_{cpd}, in this memory hierarchy? Answer in nanoseconds.

$$0.985 \times 0.5 + 0.015 \times \left(0.99995 \times 15 \times 16 + 0.00005 \times 8 \times 10^6\right) = 10.09\,\texttt{ns}$$

❑ **24.2** If the CPU performs a writing operation over a cached word, how long does this operation take? Answer in nanoseconds.

Since writing a memory location involves writing simultaneously in cache and in main memory, the required time to complete this operation is $15\,$ns.

❑ **24.3** Is there any difference when writing a memory location if the word is not cached compared to the above scenario? If so, quantify this difference.

Both scenarios are different. In case of a cache miss, a block must be read from the main memory or from the disc. In average, this operation takes the following time: $\left(0.99995 \times 15 \times 16 + 0.00005 \times 8 \times 10^6\right) = 640\,\text{ns}$. Thus, the writing operation will take $640\,\text{ns} + 15\,\text{ns}$.

Exercise 25.

The cache and main memory in a computer feature the following specifications:

- Cache access time, $t_c = 0.5\,\text{ns}$.
- Main memory access time, $t_p = 10\,\text{ns}$.
- The cache memory block contains 4 words.

It is also known:

- Each writing operation carried out by the CPU implies writing the piece of data simultaneously in the cache and in the main memory.
- Cache hit rate, A_c, is 98.5%.
- 75 out of 100 memory accesses performed by the CPU are reading operations, and the rest are writing operations.

❑ **25.1** Which is the average access time taken to perform a read operation in the aforementioned memory hierarchy? Answer in nanoseconds.

$$A_c \times t_c + (1 - A_c) \times t_p \times B = 0.985 \times 0.5 + 0.015 \times 10 \times 4 = 1.09\,\text{ns}$$

❑ **25.2** Which is the average access time in the aforementioned memory hierarchy? Answer in nanoseconds.

$$T = T_{read} + T_{write} = 0.75 \times 1.09 + 0.25 \times (1.09 + 10) = 3.59\,\text{ns}$$

Exercise 26.

In some memory hierarchies, the behaviour of writing and reading operations may be different. Writing in a hierarchy level is replicated in the next level, as it happens in write-through cache memories.[1]

In this type of cache, when a hit occurs while writing, the writing operation is carried out at the same time in the cache and in the main memory. Thus, the writing time matches the writing time of the main memory. When a cache miss occurs in a writing operation, the following strategies can be used:

[1] Write-through cache will be studied in the next unit.

1. *No write allocate*. The piece of data is written in the main memory.
2. *Write allocate*. The block containing the piece of data to be written in the main memory is copied to the cache. Then, the writing operation is carried out in the cache and in the main memory simultaneously.

It must be taken into account the fact that when using write-through writing the average access time will differ from reading to writing. It will depend on the allocation strategy. Next, an example shows this fact.

A memory system has the following characteristics:

- Cache. Type: *write through*. Average access time, t_c, to a 64-bit-wide word: 0.5 ns.
- Main memory. Average access time, t_p, to a 64-bit-wide word: 15 ns.
- Disk. Average access time, t_d, to any block from 1 byte to 10 KB: 8 ms.

It is also known:

- Cache hit rate with no write allocate, $A_{c-nwr} = 98.5\%$.
- Cache hit rate with write allocate, $A_{c-wr} = 99\%$.
- Main memory hit rate, $A_p = 99.995\%$.
- Cache block size: 16 words, 64 bits each.
- Disk block size: 4 KB.
- Memory access rate: 70% reading, 30% writing.

Taking the above information into account, answer the following questions:

❑ **26.1** What is the reading average time, tr_{cpd}, in this memory hierarchy in each configuration? Answer in nanoseconds.

(1) $0.985 \times 0.5 + 0.015 \times \left[0.99995 \times 15 \times 16 + 0.00005 \times 8 \times 10^6\right] = 10.09$ ns
(2) $0.990 \times 0.5 + 0.010 \times \left[0.99995 \times 15 \times 16 + 0.00005 \times 8 \times 10^6\right] = 6.89$ ns

❑ **26.2** What is the writing average time, tw_{cpd}, in this memory hierarchy in each configuration? Answer in nanoseconds.

(1) $tw_{cpd} = A_c \times t_p + (1 - A_c)\left[A_p \times t_p + (1 - A_p) \times (t_d + t_p)\right] =$
$0.985 \times 15 + 0.015 \times \left[0.99995 \times 15 + 0.00005 \times (8 \times 10^6 + 15)\right] = 21$ ns
(2) $tw_{cpd} = A_c \times t_p + (1 - A_c)\left[A_p \times t_p \times (B_{cp} + 1) + (1 - A_p) \times (t_d + t_p)\right] =$
$0.990 \times 15 + 0.010 \times \left[0.99995 \times 15 \times (16 + 1) + 0.00005 \times (8 \times 10^6 + 15)\right] =$
21.4 ns

❑ **26.3** Taking the answers of the two previous questions into account, does the no-write-allocate strategy provide the better performance? Why?

No. It provides a better performance for writing, but not for reading.

❑ **26.4** What is the average access time, t_{cpd}, in this memory hierarchy in each configuration? Answer in nanoseconds.

(1) $t_{cpd} = P_r \times tr_{cpd} + P_w \times tw_{cpd} = 0.7 \times 10.09 + 0.3 \times 21 = 13.36$
(2) $t_{cpd} = P_r \times tr_{cpd} + P_w \times tw_{cpd} = 0.7 \times 6.89 + 0.3 \times 21.4 = 11.24$

❑ **26.5** What configuration provides a better performance for the memory access pattern described in this exercise? Why?

```
Write allocate, since it provides the shortest average access
time.
```

Exercise 27.

There exists a factor influencing greatly the access time to a memory hierarchy together with the cache and main memory size. Which is this factor?

```
The locality of the program, which determines the hit rates in
different levels of the memory hierarchy.
```

Exercise 28.

One program manages the queries made by thousands of users simultaneously to a bank database. The memory access pattern of this type of applications is quite random, having a bad impact over the system performance. What do you think the reason is?

```
Random access patterns involve low locality of the program,
both spatial and temporal. Thus, the memory time access is neg-
atively affected.
```

Exercise 29.

Which of the following statements are TRUE? You may answer NONE if you think all of them are false.

A) The size of the block used to communicate different levels in the memory hierarchy tends to decrease in levels far from the CPU.

B) Usually, the memory organization in a hierarchy tends to focus on high speed and low cost, being the capacity a less important factor.

C) The disc size has a notable effect in the performance of a memory hierarchy.

D) Using global and local variables in a program written in C favors the locality of the program.

E) Ancient computers do not have cache memory since the speed of the main memory matched the speed of the CPU.

```
E
```