

Commencé le	mercredi 2 décembre 2020, 12:00
État	Terminé
Terminé le	mercredi 2 décembre 2020, 12:30
Temps mis	29 min 54 s
Points	13,00/16,00
Note	16,25 sur 20,00 (81%)

Question 1

Correct

Note de 1,00 sur 1,00

On my CPU, I get:

```
taskset -c 0 ./memoeff memory_bound
```

```
>> memory_bound 1 : 0.310879
```

```
>> memory_bound 8 : 0.433224 (jump of size8)
```

What does it mean?

Veillez choisir une réponse :

- ☐ a. If there is a memory jump of 8Bytes, then it is always slower.
- ☒ b. Each memory access loads a cache line, thus it takes almost the same time to use one of several values in each line. ✓
- ☐ c. It is always faster to run a loop that load less memory.

Votre réponse est correcte.

La réponse correcte est : Each memory access loads a cache line, thus it takes almost the same time to use one of several values in each line.

Question 2

Correct

Note de 1,00 sur 1,00

In memory_bound, to iterate over different "cache lines", we have to iterate each:

Veillez choisir une réponse :

- ☐ a. 8 bytes.
- ☐ b. 64 bits.
- ☒ c. 64 bytes. ✓
- ☐ d. 64 long int.

Votre réponse est correcte.

Question **3**

Correct

Note de 1,00 sur 1,00

What is the aim of the cache_k function?

Veillez choisir une réponse :

- ☐ a. It iterates and jumps every k and shows that it has no effect on the execution time.
- ☒ b. It shows that the execution time depends on the cache level that can contain the data. ✓
- ☐ c. It changes a given part of the array to show that the execution time is proportional to k.
- ☐ d. It updates the same cache line several times with different values.

Votre réponse est correcte.

La réponse correcte est : It shows that the execution time depends on the cache level that can contain the data.

Question 4

Incorrect

Note de 0,00 sur 1,00

Consider:

```
$ taskset -c 0 ./memoeff cache_k
>> cache_k 0 : 2.62968 (limit 0KB)
>> cache_k 1 : 2.87553 (limit 0KB)
>> cache_k 2 : 2.64053 (limit 0KB)
>> cache_k 3 : 2.61068 (limit 0KB)
>> cache_k 4 : 2.84341 (limit 0KB)
>> cache_k 5 : 2.73469 (limit 0KB)
>> cache_k 6 : 2.62673 (limit 0KB)
>> cache_k 7 : 2.70262 (limit 0KB)
>> cache_k 8 : 2.71228 (limit 0KB)
>> cache_k 9 : 2.69635 (limit 0KB)
>> cache_k 10 : 2.66076 (limit 1KB)
>> cache_k 11 : 2.64339 (limit 2KB)
>> cache_k 12 : 2.6423 (limit 4KB)
>> cache_k 13 : 2.67841 (limit 8KB)
>> cache_k 14 : 2.79771 (limit 16KB)
>> cache_k 15 : 4.03903 (limit 32KB)
>> cache_k 16 : 3.37497 (limit 64KB)
>> cache_k 17 : 2.6408 (limit 128KB)
>> cache_k 18 : 2.76647 (limit 256KB)
>> cache_k 19 : 4.00493 (limit 512KB)
>> cache_k 20 : 5.57972 (limit 1024KB)
>> cache_k 21 : 5.30351 (limit 2048KB)
>> cache_k 22 : 5.37049 (limit 4096KB)
>> cache_k 23 : 5.78312 (limit 8192KB)
>> cache_k 24 : 6.17291 (limit 16384KB)
>> cache_k 25 : 6.15476 (limit 32768KB)
>> cache_k 26 : 5.84435 (limit 65536KB)
>> cache_k 27 : 5.31071 (limit 131072KB)
>> cache_k 28 : 5.17969 (limit 262144KB)
>> cache_k 29 : 5.16168 (limit 524288KB)
>> cache_k 30 : 5.14543 (limit 1048576KB)
```

What are the sizes of the caches on this machine?

Veillez choisir une réponse :

- ☒ a. L1 16k, L2 64K, L3 256K. ✖
- ☐ b. L1 32k, L2 64K, L3 8192K.
- ☐ c. L1 8k, L2 16K, L3 256K.

Votre réponse est incorrecte.

La réponse correcte est : L1 32k, L2 64K, L3 8192K.

Question **5**

Correct

Note de 1,00 sur 1,00

The ilp function shows the instruction level parallelism. What does it mean in terms of performance?

Veillez choisir une réponse :

- ☐ a. It takes fewer times to compute a number if its predecessor has just been computed.
- ☐ b. A mul followed by a plus is faster than a plus follow by a mul.
- ☒ c. Independent instructions can be computed at the same time/cycle. ✓

Votre réponse est correcte.

La réponse correcte est : Independent instructions can be computed at the same time/cycle.

Question **6**

Correct

Note de 1,00 sur 1,00

The Store-to-load forwarding mechanism is a trick used by the CPU to avoid loading a value from the memory if the same memory block is move for a store. However, the CPU cannot compare the real address of load and store, but uses the last 12 bits (the part under 4096).

Here is an example of output:

```
taskset -c 0 ./memoeff aliasing4k
>> no conflict : 0.800213
>> conflict : 1.05208
```

Therefore, the function aliasing4k tries to show this effect. In the second loop, what is the expected behavior?

Veillez choisir une réponse :

- ☐ a. At each iteration, "a[i] += " will throw a load that will conflict will "b[i]" store.
- ☐ b. At each iteration, "a[i] += " will throw a load that will conflict will "b[i]" load.
- ☐ c. At each iteration, "a[i] += " will throw a store that will conflict will "b[i]" store.
- ☒ d. At each iteration, "a[i] += " will throw a store that will conflict will "b[i]" load. ✓

Votre réponse est correcte.

La réponse correcte est : At each iteration, "a[i] += " will throw a store that will conflict will "b[i]" load.

Question **7**

Correct

Note de 1,00 sur 1,00

From your implementation of the misprediction function, how would you describe the mechanism?

The CPU select an execution path...

Veillez choisir une réponse :

- ☐ a. By evaluating in advance the instruction.
- ☐ b. Always select the same path.
- ☒ c. Select the path that was previously valid. ✓

Votre réponse est correcte.

La réponse correcte est : Select the path that was previously valid.

Question **8**

Correct

Note de 1,00 sur 1,00

"False sharing" is a term used to describe the effect of modifying a part of cache line, which is also used by another thread that works on the other part.

The memory system will have to exchange this cache line to ensure memory coherency.

You will see that the variable are marked as volatile, why is that?

Veillez choisir une réponse :

- ☐ a. volatile means that the variable is used by multiple threads.
- ☒ b. volatile means that the variable cannot live in the cache. ✓
- ☐ c. volatile means that only one thread uses this variable.

Votre réponse est correcte.

La réponse correcte est : volatile means that the variable cannot live in the cache.

Question **9**

Correct

Note de 1,00 sur 1,00

The CPU likes regular memory accesses, and the prefetching function tries to show that.

Consider the output:

```
$ taskset -c 0 ./memoeff prefetching
```

```
>> regular access : 12.8107
```

```
>> irregular access : 15.3077
```

How would you explain these results?

Veillez choisir une réponse :

- ☐ a. The second part of the function does jump between values, so it is slower.
- ☒ b. The second part of the function does jump between cache line, so the CPU cannot load the cache lines in advance. ✓
- ☐ c. The second part of the function does more instructions, so it is slower.

Votre réponse est correcte.

La réponse correcte est : The second part of the function does jump between cache line, so the CPU cannot load the cache lines in advance.

Question **10**

Incorrect

Note de 0,00 sur 1,00

From the cache_effect function, how many set are there in a cache:

Veillez choisir une réponse :

- ☐ a. size of cache divide by size of a long int, for example 32KB/64bits
- ☒ b. size of cache divide by k times size of a cache line, for example 32KB/k*64B ✗
- ☐ c. size of cache divide by size of a cache line, for example 32KB/64B

Votre réponse est incorrecte.

La réponse correcte est : size of cache divide by size of a cache line, for example 32KB/64B

Question **11**

Correct

Note de 1,00 sur 1,00

If there are 4092 set in the cache, what is the jump in memory to have two address that should go in the same set?

Veillez choisir une réponse :

- ☒ a. Number of set times size of cache line, for example 4096B times 64B. ✓
- ☐ b. Number of set times size of the cache, for example 4096B times 32KB.
- ☐ c. Number of set times size divided by the size of cache line, for example 4096B divided by 64B.

Votre réponse est correcte.

La réponse correcte est : Number of set times size of cache line, for example 4096B times 64B.

Question **12**

Correct

Note de 1,00 sur 1,00

The cache_effect function is composed of three parts, which one should be slower?

Veillez choisir une réponse :

- ☐ a. 2
- ☒ b. 3 ✓
- ☐ c. 1

Votre réponse est correcte.

La réponse correcte est : 3

Question **13**

Correct

Note de 1,00 sur 1,00

What does "Temporal locality" mean?

Veillez choisir une réponse :

- ☐ a. If a data is used at 1pm one day, it is going to be used at 1pm tomorrow and the other days.
- ☒ b. If a data is used at 1pm one day, it is going to be re-used just after like 1pm + 1ns. ✓
- ☐ c. If a data is used at 1pm, it has to be placed in a specific part of the main memory.

Votre réponse est correcte.

Question **14**

Correct

Note de 1,00 sur 1,00

Among the proposed mechanisms, which one help to improve performance when the code has temporal locality?

Veillez choisir une réponse :

- ☐ a. LRU in the registers.
- ☒ b. LRU in the caches. ✓
- ☐ c. LRU in the main memory.

Votre réponse est correcte.

La réponse correcte est : LRU in the caches.

Question **15**

Correct

Note de 1,00 sur 1,00

What does "Spatial locality" mean?

Veillez choisir une réponse :

- ☐ a. If a variable "x" is loaded, then all variables of the same type are loaded.
- ☐ b. If a data at address x is used, data closed to it should not be used.
- ☒ c. If a data at address x is used, data closed to it will be quickly accessible. ✓
- ☐ d. Data that are far in memory remain far in the cache.

Votre réponse est correcte.

La réponse correcte est : If a data at address x is used, data closed to it will be quickly accessible.

Question **16**

Incorrect

Note de 0,00 sur 1,00

Among the proposed mechanisms, which one help to obtain good performance when a code has a good spatial locality?

Veillez choisir une réponse :

- ☒ a. The fact that CPU have register. ❌
- ☐ b. The fact that cache lines are of size 64B.
- ☐ c. The fact that energy consumption changes with data movement.
- ☐ d. The fact that CPU has DRAM.

Votre réponse est incorrecte.

La réponse correcte est : The fact that cache lines are of size 64B.

◀ Subject

Aller à...



subject ▶