christian.hakert [☺] tu-dortmund.de
marcel.ebbrecht [☺] tu-dortmund.de

Exercises for
Computer Architecture
Summer Semester 2020

Exercise Sheet 3

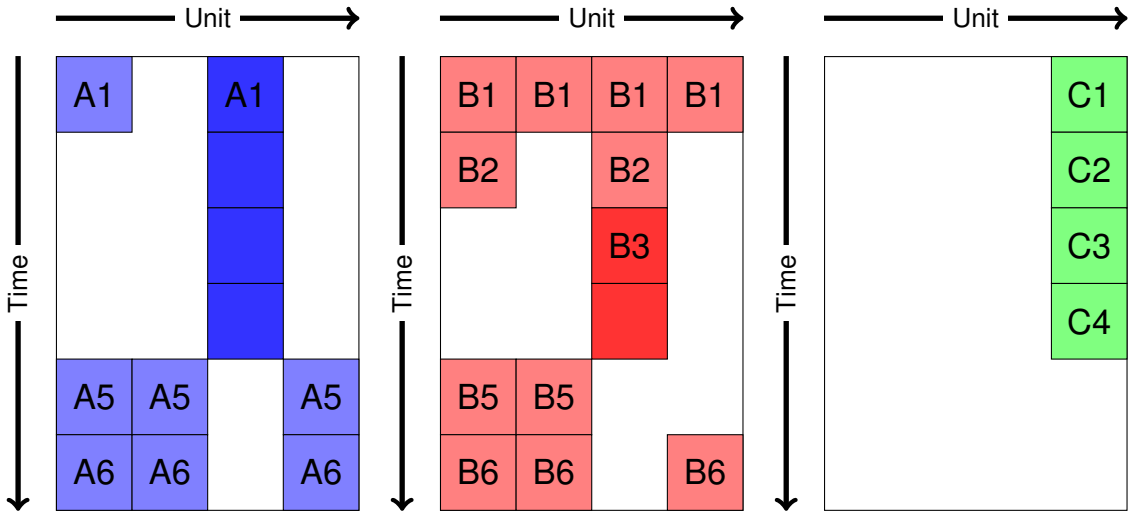
Discussion starts from Monday, May 11, 2020

3.1 Multithreading

Given a system with one processor and dynamic scheduling, which contains two integer units (1 cycle latency), one load/store unit (1-4 cycles latency, depending on the cache behaviour) and one floating-point unit (1 cycle latency). Additionally, a L1-Cache and a L2-Cache are present with following latencies:

L1-Access	L2-Access	Latency (Cycles)
Hit	-	1
Miss	Hit	2
Miss	Miss	4

Both cache allow *Hit-Under-1-Miss*, i.e. while one miss is processed, other hits can be processed. Another muss has to be canceled and repeated by the processor.



(a) Thread A (Priority 0) (b) Thread B (Priority 1) (c) Thread C (Priority 2)

Abbildung 1: Execution behaviour of 3 threads on the given processor

For this system, the execution behaviour of 3 programs (A-C) in isolation are depicted in Figure 1. Operations which consume more than one cycle are indicated with a darker color. The priority of all threads is given (only required for CGMT and SMT), lower numbers imply higher priorities.

a) Draw the execution under *Coarse-Grained Multithreading (CGMT)*, *Fine-Grained Multithreading (FGMT)* and *Simultaneous Multithreading (SMT)* for the given applications running in in one thread at the same time on the given system. Refer to the notation of Figure 1, units are grouped horizontal, the time is grouped vertical. The *Context-Switch-Overhead* is 1 cycle for CGMT and 0 cycles for the other methods.

You can assume, that the memory accesses of all threads target disjoint memory regions, such that no cache interferences can occur. The used scheduler is the ASAP (As Soon As Possible) scheduler. If not all instructions of one time unit can be started at the same time, the scheduler executes the remaining instructions as soon as possible. Further assume, that all instuctions of one time unit have to complete before instructions from the next time unit of the same thread can start.

b) What are the hardware requirements for these three execution models?

3.2 Coarse-grained Multithreading

For the execution of the instructions of all threads of one application, the following distribution is given:

- 40% of all cycles are used for execution of the instructions in the processor.
- 30% of all cycles are spent on waiting for L1-Cache-Misses (L2-Hits). The handling of L1-Misses takes 10 cycles.
- 30% of all cycles are spent on waiting for L2-Cache-Misses, which take 30 cycles.

In average, a Cache-Miss happens after 10 cycles in each thread. Answer the following questions under the assumption, that sufficient threads are provided by the hardware, as well as by the application.

a) How much is the maximal utilization, if one context-switch takes 5 cycles?

b) What is the maximal latency for a context-switch, if the utilization should be at least 50%?

3.3 Parallel Text Processing with OpenMP

a) Write a C++ Programm, which generates 10MB of random text. Use the following function for the generation:

```
void initText( string &s ) {
static const char alpha[] =
"abcdefghijklmnopqrstuvwxyz";

for ( int i = 0; i < s.length(); i++ ) {
s[i] = alpha[rand() % (sizeof(alpha) - 1)];
}
}
```

b) Implement a simple text search, which searches a fixed string S within the random generated text T. This should be done, by comparing the entire string S with each suffix from T. Measure the execution time with the C function

Implementieren Sie eine Zeitmessung für die Suchfunktion mithilfe der C-Funktion clock. Pay attention to omit all I/O operations (printf, ...) during the string search.

c) Try different possibilities to parallelize your program from b) with the help of OpenMP-Pragmas. Measure the execution time again and determine the best way of parallelization.

d) Extend your solution from c), such that the number of found texts is counted during the search. Use the synchronization mechanisms of OpenMP to maintain the correctness of the counted amount.

e) Cyn you maintain the correctness of d) also with a barrier?

General Information: Further information can be found under hhttps://ls12-www.cs.tu-dortmund.de/daes/de/lehre/lehrveranstaltungen/summersemester-2020/rechnerarchitektur-deutsch.html. Submitting solutions to the exercise sheets is not required.