Concurrency — Exercise 3
Locking 1

Prof. Dr. Oliver Haase

Problem 1

What is the *double-checked locking* anti-pattern? Why is it an anti-pattern?

Problem 2

Complement the `Downloader` class from the lecture with a `ProgressListener` implementation and a `main` method that lead to a deadlock. Draw a sketch that illustrates the deadlock situation. Your sketch should contain the threads, the locks, as well as which threads hold and try to acquire which locks at the time of deadlocking.

Problem 3

Write a short test program to measure the performance overhead for uncontended locks. More concretely, implement a synchronized as well as an unsynchronized version of a simple method, execute both versions single-threaded many times and measure and compare the execution times.

Problem 4

Extend your implementation of example 3 from the lecture, the factorizer service that writes and uses the cache, with appropriate locking.

Problem 5

Now modify your solution to problem 4 by making an atomic snapshot of the state variables and then operating on this snapshot rather than on the shared data.

Have fun and good luck!