

Concurrent Programming – Fall/Winter 2019/20

Instructor: Michael Mäder

michael.maeder@hefr.ch

Recommended Books:

The JR Programming Language, by Ronald Olsson and Aaron Keen, ISBN: 1-4020-8085-9

Concurrent Programming, Principles and Practice, by Gregory R. Andrews, ISBN: 0-8053-0086-4

Principles of Concurrent and Distributed Programming, by Morchedai Ben-Ari, ISBN: 032131283X

Tentative Plan for concurrent programming 1, Fall semester 2019/2020:

Week	Subject
A1	Motivation for concurrent programming; critical section problem; introduction to JR
A2	Techniques for avoiding interference; Making sure your system is set up
A3	Semaphores: mutual exclusion; Producers and Consumers; first lab description
A4	Barrier synchronization: the technique of passing the baton; Selective mutual exclusion: Readers and Writers
A5	Dining Philosophers; Threads and Semaphores in Java; volatile
A6	synchronized - Atomic Integer, Locks
A7	Sample Exam Questions - Lab
A8	Exam
A9	Monitors – notation, signaling disciplines – preprocessor with jr
A10	Synchronizing techniques – priority; implementing monitors with semaphores
A11	Lab
A12	Sleeping Barber Problem - Monitors & Java: pre 1.5 versions
A13	Lab
A14	Monitors – the pthread library
A15	Lab
A16	Exam

Grading – 2 Exams during the semester, at least 4 lab programs to turn in. In addition, there is an oral final exam during the first full week of February. The grade will be computed as follows:

$$\frac{\text{oralexam} + \frac{\text{exam}_1 + \text{exam}_2}{2} + \frac{\text{lab}_1 + \text{lab}_2 + \dots + \text{lab}_n}{n}}{2}$$