Course syllabus

Flertrådad programmering
Concurrent Programming

EDAP10, 7,5 credits, A (Second Cycle)

Valid for: 2023/24
Faculty: Faculty of Engineering, LTH
Decided by: PLED C/D
Date of Decision: 2023-04-18

General Information

Main field: Technology.
Compulsory for: D3
Elective for: BME4-sbh, C4-pv, E4-pv, F4, F4-pv, M4-me, N4, Pi4-pv
Language of instruction: The course will be given in Swedish

Aim

Concurrent programs combine parallel, asynchronous activities, frequently interacting with a physical environment. Applications include control systems, network communication, and interactive systems. The course aims to give students ability and skills to design robust, efficient, and well-structured concurrent programs.

Learning outcomes

Knowledge and understanding
For a passing grade the student must

- be familiar with and be able to describe how threads execute and are synchronized in a concurrent program
- be familiar with causes for race conditions, and techniques for avoiding them
- be familiar with causes for deadlock, and techniques for avoiding it

Competences and skills
For a passing grade the student must

- be able to design concurrent reactive software based on application requirements and prescribed interfaces to the environment
- be able to implement and debug multi-threaded software

- be able to perform simple deadlock analysis for a small system of threads and resources

Judgement and approach
For a passing grade the student must

- be able to compare and discuss design and implementation options for concurrent programs
- be able to make trade-offs between performance and implementation complexity in concurrent programs

Contents

- Basic concepts: concurrent activities, busy-wait and polling, synchronization and communication, atomic operations such as test-and-set, and mutual exclusion.
- Synchronization and signaling in shared-memory applications: locks, monitors.
- Synchronization and signaling in message-passing applications.
- Concurrency in interactive systems with callbacks.
- Analysis of deadlock and livelock.
- Central aspects of the Java concurrent package, such as locks, semaphores, thread pools, atomics, and blocking queues.
- Overview of real-time operating systems, scheduling, multicore hardware, lock-free algorithms. Brief introduction to more advanced frameworks in the Java concurrent package.

Examination details

Grading scale: TH - (U,3,4,5) - (Fail, Three, Four, Five)
Assessment: Compulsory lab work and written examination. The final grade of the course is based on the result of the written examination.

The examiner, in consultation with Disability Support Services, may deviate from the regular form of examination in order to provide a permanently disabled student with a form of examination equivalent to that of a student without a disability.

Parts
Code: 0119. **Name:** Written Examination in Concurrent Programming.
Credits: 3. **Grading scale:** TH. **Assessment:** Written examination. The final grade of the course is based on the result of the written exam.
Code: 0219. **Name:** Compulsory Course Items.
Credits: 4,5. **Grading scale:** UG. **Assessment:** For a passing grade, the laboratory work must be completed. **Contents:** Compulsory laboratory work.

Admission

Admission requirements:

- EDAA01 Programming - Second Course or EDAA30 Programming in Java - Second Course

The number of participants is limited to: No
The course overlaps following course/s: EDA698, EDA040, EDAF85, EDAF55

Reading list

- Brian Goetz: Java Concurrency in Practice. Addison-Wesley, 2006, ISBN: 978-0321349606.

- Additional materials provided by the department.

Contact and other information

Course coordinator: Patrik Persson, patrik.persson@cs.lth.se
Course homepage: https://cs.lth.se/edap10/