Outline

1 Syllabus
2 Schedule
3 Literature
4 Overview
Course Syllabus – Learning outcomes

The students shall after the course be able to:

- Explain in detail how techniques such as Semaphores, Monitors, Message passing and busy wait, work.
- Reason about how the techniques can be used to deal with concurrency in computer systems, including judging their usability in different situations.
- Account for problems and possibilities with concurrency in computer systems.
In swedish – Mål

- Till kursbeskrivningen...för ambitiösa studenter...
Examination

3 lab sessions
Instructions given 0-2 days before

3 “tick” sessions
Instructions given 5-7 days before
Problem solving sessions which are student-led.

- ✔️ for correct exercise
- ✗ for wrong exercise
- We only count the ✔️

Grades...

3: 50% of ✔️ + 2 labs
4: 65% of ✔️ + 2 labs
5: 80% of ✔️ + 3 labs
U: otherwise
How to get ✔️?

You must demonstrate:

- **honest effort** to prepare the problem
- be able to **lead a classroom discussion** to a satisfactory solution
Outline

1 Syllabus
2 Schedule
3 Literature
4 Overview
Preliminary Schedule
Outline

1. Syllabus
2. Schedule
3. Literature
4. Overview
Literature

Gregory Andrews. 
*Foundations of Multithreaded, Parallel and Distributed Programming.* 
Addison-Wesley, 1999

(ISBN: 0-201-35752-6)
Gadi Taubenfeld. 
*Synchronization algorithms and Concurrent Programming.*
Pearson, 2006
(ISBN: 0-131-97259-6)
Brian Goetz. 
*Java Concurrency in practice.*
Addison-Wesley, 2006
William Stallings.  
*Operating systems, Internals and Design Principles, 5th edition.*  
Addison-Wesley, 2006  
(ISBN: 0-131-27837-1)
Outline

1. Syllabus
2. Schedule
3. Literature
4. Overview
Overview

- Introduction (course details, etc...)
- Classical paradigms + Process/Threads in OSes
- Hardware
- Shared memory programming
- Locks
- Barriers
- Semaphores
- Monitors
- Message Passing
- Implementation in OSes
- Extras
- Summary