Connections

- Applications
- Operating systems
- Programming languages
- Concurrent Programming (Rio)
  - synchronization
  - communication
  - co-operation
- Distributed systems
- Performance
- Computer Architecture

Motivation

- To know and understand …
  - Why concurrency is important
  - What are the concurrency problems in your systems
  - How concurrency problems are usually solved
  - What are the usual tools for solving concurrency problems
  - How concurrency problems may be solved at different system levels
    - HW, OS, progr. language library, application

Learning Goals

- Concurrency and problems caused by concurrency
  - Basics, fundamental ideas, background
- Fundamental concepts and models in concurrency
  - Main emphasis is here!
  - E.g., “can explain special features of semaphores and monitors and use them properly in applications”, or “can explain how deadlocks can be prevented”.
- Concurrent programming in distributed systems
  - Get taste of it, more in later courses
- Concurrent programming in practice
  - Discussed in lectures, practice in project

Course Connections

- Computer Organization I
  - Tietokoneen toiminta
- Concurrent Programming
  - Rinnakkaisohjelmointi
- Operating Systems
  - Käyttöjärjestelmät
- Distributed Systems
  - Hajautetut järjestelmät
- Software Design (Java)
  - Ohjelmointitekniikka (Java)
- Network Programming
  - Verkkoosovellusten toteuttaminen
- Linux System Admin
  - Linux-ylläpito
- Intro to Specification and Verif.
  - Spesifioinnin ja verif. perusteet
- Huawei
  - Vensim
- Java
  - Java
- Funipraktikum
  - Funipraktikum
- Java
  - Java
What Good is Concurrent Programming for?

- All computer systems are inherently concurrent – one must understand concurrency to understand computer system operation
- Concurrency allows huge speedups for properly designed systems
- Concurrency causes complex problems that are not easy to solve without good understanding of concurrency
  - Should I use threads in my Java application or not?
  - Would it be better to use locks, semaphores, monitors or transactional memory to solve synchronization and communication problems in my application?
  - Why doesn’t it do what I thought it would do?
  - How can I show my boss that it really works?

What is Not Covered?

- How to write efficient code for multicore architectures?
- What types of applications are suitable for multicore architectures?
- What programming paradigms exist to write efficient code for multicore architectures?
- How to write multicores code so that it would run in many multicore architectures?
- How to best utilize multicore GPU in your application?
  - CUDA - Compute Unified Device Architecture
  - How to partition your solution to multicore CPU & GPU?
- Programming distributed applications

Learning Methods

- Lecture notes
  - Not perfect for self-study – use with text book
- Summary lectures & discussions
- BACI – Ben-Ari Concurrency Interpreter
- Practice problems
- Homeworks
- Project
- Group meetings (practice sessions)
- Learning diary (elective)
- Course exam

Summary Lecture and Discussion

- Short summary lecture
  - Some of the slides for 2008 lectures
  - Assume: students have read the text book in advance
  - Basic knowledge of today's topic
- Discussion (Socratic discussion)
  - Goal: deeper understanding of today’s topic
  - Students discuss
    - Starting points: questions, claims
    - Students, moderator
    - Goal: answer the question, verify the truth of the claim
  - Moderator may intervene to keep discussion on track

Java

- How to use Java for Concurrent Programming?
  - Threads
  - Messages
  - Semaphores
  - Monitors
- Lectures, projects
Study Circles

• Group work, team work
  – It is better to study in a team than alone
  – Peer student support
  – Study circles formed in the first group meeting

• Student centered learning
  – The student has responsibility on learning
  – Instructor facilitated learning
  – Instructors give good environment for learning

• Three types of team work
  – Solving homework problems independently and then discussing them in study circles and in group meetings
  – Projects
  – Any other co-operative work for this course

Creation of Study Circles

• Possibly the largest problem in study circle courses
  – “Ville promised, but did not do and he was not accessible. And then Maija did most of it. This is not right! Boohoo! 😞”

• Study circles are formed in the 1st group meeting
  – Goals should be similar
  – Think about your goals before the 1st group meeting
  – Discuss and agree on common goals before agreeing on forming a study circle
  – Finally, sign the “Study Circle Contract”

  • Keep up with your agreement
  – Inform the study circle immediately, if you will not continue
  – Get quickly rid of peer students who do not work as agreed on

Practice Problems

• Practice problems
  – Self evaluation
    • Do them only after you think you know the material
    • Do I understand it now?
    – Use does not directly affect your grade
    • No bookkeeping on material use
    • No credit toward course grade

  – More practice problems are made in the project

Homework Problems

• Normal homework problems
  – Learning happens when you solve the problems and discuss them
  – Study topic area first before trying out the problems
  – Reading a complete solution or giving one to peer student is wasting a good problem!

• Homework problems are discussed at
  – Study circle own meetings before group meetings
  – Group meetings with peer students at the table

  • Affects your grade
  – You get homework points (hwp, lhp) for completed problems
  – Only for those present in group meeting

Group Meetings

• Mark down the problems you have completed
• Assistant will organize you in tables
  – Each table should have a student solution for each problem
  – One or more tables may be in English in English speaking practice session

• Discuss all problems in your own table
  – You should have at least tried to solve all problems beforehand
  – Correct solutions available, consult assistant if needed
  – No presentations in front of class

• Discuss additional topics given in solutions paper
  – Open ended discussions, no “correct” answers given

• Common discussion of selected problems/solutions
• Advice students in other tables, if there is time for it

Project 1

• Project (A&B or A&C) with team work
  A. Come up with a new practice problems
  B. Do a small project in Java from text book
  C. Write a better guide on how concurrent programming is done with Java

• Goal is deeper understanding on some topics
  – Other course components may be needed as background knowledge

• Affects your grade
  – Instructor evaluates the report
  – Grade (1-5 pp) depending on the quality of work

  • Participation points (max ±2 pp) based on your participation
  – Study circle determines this part!
Learning Diary (Extra Project 2)

• Evaluate and reflect
  – Cover all learning events (lecture, homework, practice meeting, study circle meeting, etc.)
  – What did you do, observe, learn, and feel

• Affects your grade
  – Same way as other projects
  – Points (pp) are completely extra, and
  you can get an excellent grade also without this project
  – With learning diary you probably get a better grade!

Studying for This Course

• Study weekly topics
  – Read the text book the same topics, with different approach
  – Use lecture slides if you find them useful
  – Attend summary lectures and actively participate discussions
• Check your learning with self evaluation
  – Do practice problems and homework
• Participate in study circle
  – Discuss homework
  – Weekly group meeting
  – Continue projects
  – Study circle meet face-to-face or in the web
• Finish project 1 in time
• Study for exam
  – Take course exam
  – Do extra project
  – Evaluate, think, do, reflect

Evaluation

• Self evaluation
  – Do practice problems after each topic
  – Does not affect your grade
  – Do homeworks each week
  – Do I understand or not?
  – What is there still to learn and how do I do it??
  – Affects your grade
• Course exam
  – Gives a fixed deadline for learning
  – Covers all topics
  – Topics learned in independent study as well as in study circles using various learning methods
  – Examines learning
  – Most of the grade based on this
  – Must reach certain level (50%) to pass the course

Grading

• Good work is awarded
• Diligence and knowledge is awarded
• Course component maximum grade points

<table>
<thead>
<tr>
<th>Component</th>
<th>Max Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homeworks (min 1p)</td>
<td>6 p</td>
</tr>
<tr>
<td>Study circle project (min 1p)</td>
<td>10 p</td>
</tr>
<tr>
<td>Course exam (min 22 p)</td>
<td>44 p</td>
</tr>
<tr>
<td>Total (min 30)</td>
<td>60 p</td>
</tr>
</tbody>
</table>

Course contents

• Lecture 0: Admin
• Lecture 1: Concurrency
• Lecture 2: Concurrency at Prog. Lang. Level
• Lecture 3: Critical Section Problem
• Lecture 4: Verifying Concurrent Programs
• Lecture 5: Deadlocks
• Lecture 6: OS Support for Conc: Semaphores
• Lecture 7: More on semaphores
• Lecture 8: Progr. Lang. Support for Conc: Monitors
• Lecture 9: Concurrency Control in Distr. Environment
• Lecture 10: Crit. Sections in Distributed Environment
• Lecture 11: Practical Examples on Concurrency Control
• Lecture 12: Current Research, Course Summary
• Project: Java programming

Motto

• “It is not good exercise, if you do not sweat”
• This is not a marathon!
• Study-week approach
  – Altogether some 12 h / week
  + exam preparation + exam
  – Total some 80 h / 2 sw course (2 work weeks)
• Top-down approach
  5 yrs / 300 cu = 1 yr / 60 cu = 1600 h / 60 cu
  = 26.67 h / 1 cu = 107 h / 4 cu
  – Total some 107 h / 4 cu course
Nina Aremo study 2007

- Questionnaire study, weekly basis
  - 40/121 student participated (14 women, 27 CS majors)
- Course workload opinion: 50% ok, 50% too much
  - Average weekly work hours: 7, 8, 10, 9, 10, 11 (compared to 12)
  - Total work hours: most 50-70, max 136 (compared to 80 or 107)
- Comments
  - "Need model solutions to homework problems"
  - "Lecturer did not take questions"
  - "Difficult to do well, if you do not attend lectures and practice sessions"
  - "Web pages are incoherent"
  - "Study circles would need more guidance"
  - "Got real busy at end because of so many other courses"

Summary

- Course administration
- Course components and learning methods
- BACI simulator
- Lecture format
- Projects