Concurrent Systems: Concurrent Programming

What did you learn?

- Basic notions processes/threads, synchronization, monitors
- Awareness of pitfalls
- Application in multithreaded GUIs/servers
- Practical experience with Java Threads ($\sim$ C#, .NET)
- Should be able to use Pthreads, Win32, small kernel APIs
- Ready for *multi-cores*

Where can you learn more?

- [Goetz et al.]: *Java Concurrency in Practice*
- Parallel Computation: [Andrews III], 02614 High-perf. Comp. (E3B)
- Real-time systems: 02223 Fund.mod. of Embedded Systems
  02225 R-T Systems (F2B)
  Robotics (DTU Electro)

Concurrent Systems: Concurrency Theory

What did you learn?

- Models of concurrency: Petri-nets, interleaving of atomic actions
- Basic properties: Invariants
- Knowledge of property languages, e.g. Temporal Logic
- Knowledge of modelling languages: CSP
- Knowledge of verification tools: SPIN, (UPPAAL, . . .)

Where can you learn more?

- CSP: 02221/2 Distributed Systems (F1)
  02263 Formal Aspects of SE (F4B)
- Verification tools: 02224 Real-Time Systems (E2B)
  02246 Process modelling and validation

Concurrent Systems: Operating Systems

What did you learn?

- Knowledge about notions of processes/threads
- Knowledge of basic principles of process and memory management
- Should be able to understand a small kernel
- **Not** quite ready for Linux kernel-hacking

Where can you learn more?

- Read a standard book on OS [Tanenbaum, etc.]
- 02345 Data Security (E5)
- 02337 Mainframe - z/OS (E5B)

Concurrent Systems: Network Programming

What did you learn?

- Basic knowledge about network and protocol notions
- Knowledge about application protocols
- Experience with *socket programming* and RMI
- Can build simple client/server-based applications

Where can you learn more?

- Read a standard book on Networks [Tanenbaum, etc.]
- 02222 Distributed Systems (F1)
- 34341 Advanced Data Communication (F1B)

Concurrent Systems: Exam Fall 2008

Form

- **4 hours** — accounts for about 50 % of the overall mark
- 4-5 problems within concurrent prog. and network concepts
- Small and larger questions
- Coding of smaller program parts using pseudo-code (not Java)

Hints

- Preparation: Do exercise classes and home works (again?)
- Do not repeat problem text
- Use of pencil is acceptable
- Use time proportional to weight of problem
- Give brief justifications of your answers
- If in doubt — write down your understanding

Read the questions carefully!

Don't panic!