

4

Systems, Networks & Concurrency 2020

Organization & Contents

Uwe R. Zimmer - The Australian National University



3

Organization & Contents

what is offered here?

Fundamentals & Overview

as well as perspectives, paths, methods, implementations, and open questions

informal about

Concurrent & Distributed Systems



2

Organization & Contents

who could be interested in this?

anybody who ...

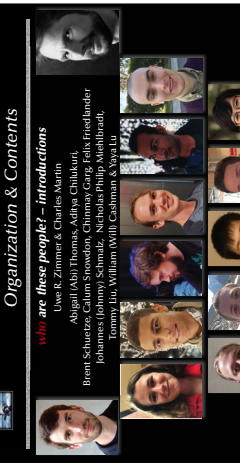
- ... wants to work with real-world scale computer systems
- ... would like to learn how to analyse and design operational and robust systems
- ... would like to understand more about the existing trade-off between theory, the real-world, traditions, and pragmatism in computer science
- ... would like to understand why concurrent systems are an essential basis for most contemporary devices and systems

5

Organization & Contents

who are these people? - introductions

Uwe R. Zimmer & Charles Martin
 Abigail (Abi) Thomas, Aditya Chikuri,
 Brent Schuetze, Calum Snowdon, Chimay Garg, Felix Friedlander,
 Johannes (Johny) Schrade, Nicholas Philip Athanassiadi,
 Zeynep Aydin, William Will, Y. Celina



6

Organization & Contents

how will this all be done?

Lectures:

- 2x 1.5 hours lectures per week ... all the nice stuff
- Tuesday 12:00 & Friday 11:00 (all live on-line)

Laboratories:

- 3 hours per week ... all the rough and action stuff
- time slots on our web-site
- enrolment: <https://cs.anu.edu.au/students/> (open since last Monday, more slots (today)

Resources:

- introduced in the lectures and collected on the course page: <https://cs.anu.edu.au/course/comp230/> ... as well as schedules, slides, sources, links to forums etc. (p. ... keep an eye on this page!)

Assessment (for discussion):

- Exam at the end of the course (60%)
- Assignments (15% + 15%)
- plus two assignments (15% + 15%)
- plus one mid-semester exam (15%)

7

Organization & Contents

Text book for the course

Ilseu, A. (Ed.)
 M. Ben-Ari
 Principles of Concurrent and Distributed Programming
 2006, second edition, Prentice-Hall, ISBN 0-13-110213-X

Many algorithms and concepts for the course are in there — *but not all!*

References for specific aspects of the course are provided during the course and are found on our web-site.

8

Organization & Contents

Topics

Language refresher [3]

1. Concurrency [3]
2. Mutual exclusion [2]
3. Communication & Synchronization [4]
4. Non-determinism [2]
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

9

Organization & Contents

Topics

- 1.1. Forms of concurrency [1]
- 1.2. Coupled dynamical systems
- 1.3. Models and terminology [1]
- 1.4. Abstractions
- 1.5. Interleaving
- 1.6. Atomicity
- 1.7. Consistent and distributed shared memory
- 1.8. Basic definitions
- 1.9. Process states
- 1.10. Implementations
2. Mutual exclusion [2]
3. Communication & Synchronization [4]
4. Non-determinism [2]
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

10

Organization & Contents

Topics

1. Concurrency [3]
2. Mutual exclusion [2]
3. Communication & Synchronization [4]
4. Non-determinism [2]
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

2.1. Shared variables [1]

- failure possibilities
- Dekker's algorithm

2.2. by test-and-set hardware support [1,2]

2.3. Mutual hardware support

2.4. Implementations [1,2]

- OS semaphores

11

Organization & Contents

Topics

- 3.1. Shared memory synchronization [2]
- Semaphores
- Cond. variables
- Conditional critical regions
- Protected objects
- Message passing [2]
- Asynchrony?
- Remote invocation (read/write/compare)
- Remote procedure call
- Addressing
4. Non-determinism [2]
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

12

Organization & Contents

Topics

- 4.1. Concurrency under non-determinism [1]
- Forms of non-determinism
- Non-determinism in distributed systems
- Consistency correctness plus non-determinism
- 4.2. Subject matters [1]
- Forms of non-deterministic message reception
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

13

Organization & Contents

Topics

- 5.1. Data parallelism
- Vectorization
- Reduction
- General data parallelism
- 5.2. Examples
- Image processing
- Cellular automata
6. Scheduling [2]
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

14

Organization & Contents

Topics

- 6.1. Problem definition and design space [1]
- Addressing: an address/solved by scheduling?
- 6.2. Basic scheduling method [1]
- Assumptions for basic scheduling
- Basic methods
- 6.3. Problem definition and design space [1]
- Addressing: an address/solved by scheduling?
- 6.4. Assumptions for basic scheduling
- Basic methods
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

15

Organization & Contents

Topics

- 7.1. Safety properties
- Essential time-independent safety properties
- 7.2. Reachability in fairness
- Classification of fairness
- 7.3. Deadlocks
- Detection
- Avoidance
- 7.4. Failure modes
- Prevention (s recovery)
- 7.5. Important & atomic operations
- Definitions
8. Distributed systems [4]
9. Architectures [1]

16

Organization & Contents

Topics

- 8.1. Networks [1]
- OSI model
- Network implementations
- 8.2. Global time [1]
- Synchronised clocks
- 8.3. Distributed states [1]
- Consistency
- Snapshots
- Termination
- 8.4. Distributed communication [1]
- Name spaces
- Multi-casts
- Elections
- Network identification
- 8.5. Distributed safety detection
- Distributed deadlock detection
- 8.6. Forms of distributed redundancy [1]
- computation
- memory
- operations
- 8.7. Transactions [2]
- Two-phase locking
9. Architectures [1]

17

Organization & Contents

Topics

1. Concurrency [3]
2. Mutual exclusion [2]
3. Condition synchronization [4]
4. Non-determinism [2]
5. Data Parallelism [1]
6. Scheduling [2]
7. Safety and liveness [3]
8. Distributed systems [4]
9. Architectures [1]

Organization & Contents

Topics

1. Concurrency [3]
2. Mutual exclusion [2]
 - CPU architecture
 - registers and address
3. Condition synchronization [4]
4. Non-determinism [2]
5. Data Parallelism [1]
 - Chapel
 - OpenMP
 - Rust
 - Julia
 - C++
7. Safety and liveness [2]
8. Distributed systems [4]
9. Architectures [1]

Organization & Contents

24 Lectures

1. Introduction [1]
 - 1.1. Motivation
 - 1.2. What is parallelism?
 - 1.3. Parallelism in hardware
 - 1.4. Parallelism in software
2. Parallelism in hardware [1]
 - 2.1. Parallelism in hardware
 - 2.2. Parallelism in hardware
 - 2.3. Parallelism in hardware
3. Parallelism in software [1]
 - 3.1. Parallelism in software
 - 3.2. Parallelism in software
 - 3.3. Parallelism in software
4. Parallelism in software [1]
 - 4.1. Parallelism in software
 - 4.2. Parallelism in software
 - 4.3. Parallelism in software
5. Parallelism in software [1]
 - 5.1. Parallelism in software
 - 5.2. Parallelism in software
 - 5.3. Parallelism in software
6. Parallelism in software [1]
 - 6.1. Parallelism in software
 - 6.2. Parallelism in software
 - 6.3. Parallelism in software
7. Parallelism in software [1]
 - 7.1. Parallelism in software
 - 7.2. Parallelism in software
 - 7.3. Parallelism in software
8. Parallelism in software [1]
 - 8.1. Parallelism in software
 - 8.2. Parallelism in software
 - 8.3. Parallelism in software
9. Parallelism in software [1]
 - 9.1. Parallelism in software
 - 9.2. Parallelism in software
 - 9.3. Parallelism in software
10. Parallelism in software [1]
 - 10.1. Parallelism in software
 - 10.2. Parallelism in software
 - 10.3. Parallelism in software
11. Parallelism in software [1]
 - 11.1. Parallelism in software
 - 11.2. Parallelism in software
 - 11.3. Parallelism in software
12. Parallelism in software [1]
 - 12.1. Parallelism in software
 - 12.2. Parallelism in software
 - 12.3. Parallelism in software
13. Parallelism in software [1]
 - 13.1. Parallelism in software
 - 13.2. Parallelism in software
 - 13.3. Parallelism in software
14. Parallelism in software [1]
 - 14.1. Parallelism in software
 - 14.2. Parallelism in software
 - 14.3. Parallelism in software
15. Parallelism in software [1]
 - 15.1. Parallelism in software
 - 15.2. Parallelism in software
 - 15.3. Parallelism in software
16. Parallelism in software [1]
 - 16.1. Parallelism in software
 - 16.2. Parallelism in software
 - 16.3. Parallelism in software
17. Parallelism in software [1]
 - 17.1. Parallelism in software
 - 17.2. Parallelism in software
 - 17.3. Parallelism in software
18. Parallelism in software [1]
 - 18.1. Parallelism in software
 - 18.2. Parallelism in software
 - 18.3. Parallelism in software
19. Parallelism in software [1]
 - 19.1. Parallelism in software
 - 19.2. Parallelism in software
 - 19.3. Parallelism in software
20. Parallelism in software [1]
 - 20.1. Parallelism in software
 - 20.2. Parallelism in software
 - 20.3. Parallelism in software
21. Parallelism in software [1]
 - 21.1. Parallelism in software
 - 21.2. Parallelism in software
 - 21.3. Parallelism in software
22. Parallelism in software [1]
 - 22.1. Parallelism in software
 - 22.2. Parallelism in software
 - 22.3. Parallelism in software
23. Parallelism in software [1]
 - 23.1. Parallelism in software
 - 23.2. Parallelism in software
 - 23.3. Parallelism in software
24. Parallelism in software [1]
 - 24.1. Parallelism in software
 - 24.2. Parallelism in software
 - 24.3. Parallelism in software

Organization & Contents

Laboratories & Assignments

1. Introduction [1]
 - 1.1. Introduction
 - 1.2. Introduction
 - 1.3. Introduction
2. Introduction [1]
 - 2.1. Introduction
 - 2.2. Introduction
 - 2.3. Introduction
3. Introduction [1]
 - 3.1. Introduction
 - 3.2. Introduction
 - 3.3. Introduction
4. Introduction [1]
 - 4.1. Introduction
 - 4.2. Introduction
 - 4.3. Introduction
5. Introduction [1]
 - 5.1. Introduction
 - 5.2. Introduction
 - 5.3. Introduction
6. Introduction [1]
 - 6.1. Introduction
 - 6.2. Introduction
 - 6.3. Introduction
7. Introduction [1]
 - 7.1. Introduction
 - 7.2. Introduction
 - 7.3. Introduction
8. Introduction [1]
 - 8.1. Introduction
 - 8.2. Introduction
 - 8.3. Introduction
9. Introduction [1]
 - 9.1. Introduction
 - 9.2. Introduction
 - 9.3. Introduction
10. Introduction [1]
 - 10.1. Introduction
 - 10.2. Introduction
 - 10.3. Introduction
11. Introduction [1]
 - 11.1. Introduction
 - 11.2. Introduction
 - 11.3. Introduction
12. Introduction [1]
 - 12.1. Introduction
 - 12.2. Introduction
 - 12.3. Introduction
13. Introduction [1]
 - 13.1. Introduction
 - 13.2. Introduction
 - 13.3. Introduction
14. Introduction [1]
 - 14.1. Introduction
 - 14.2. Introduction
 - 14.3. Introduction
15. Introduction [1]
 - 15.1. Introduction
 - 15.2. Introduction
 - 15.3. Introduction
16. Introduction [1]
 - 16.1. Introduction
 - 16.2. Introduction
 - 16.3. Introduction
17. Introduction [1]
 - 17.1. Introduction
 - 17.2. Introduction
 - 17.3. Introduction
18. Introduction [1]
 - 18.1. Introduction
 - 18.2. Introduction
 - 18.3. Introduction
19. Introduction [1]
 - 19.1. Introduction
 - 19.2. Introduction
 - 19.3. Introduction
20. Introduction [1]
 - 20.1. Introduction
 - 20.2. Introduction
 - 20.3. Introduction
21. Introduction [1]
 - 21.1. Introduction
 - 21.2. Introduction
 - 21.3. Introduction
22. Introduction [1]
 - 22.1. Introduction
 - 22.2. Introduction
 - 22.3. Introduction
23. Introduction [1]
 - 23.1. Introduction
 - 23.2. Introduction
 - 23.3. Introduction
24. Introduction [1]
 - 24.1. Introduction
 - 24.2. Introduction
 - 24.3. Introduction