Parallel Computing
„Einführung in paralleles Rechnen“

Welcome!

Intro, Remarks

Q&A

Lecturers:
- Prof. Dr. Jesper Larsson Träff
- Assoc. Prof. Dr. Sascha Hunold

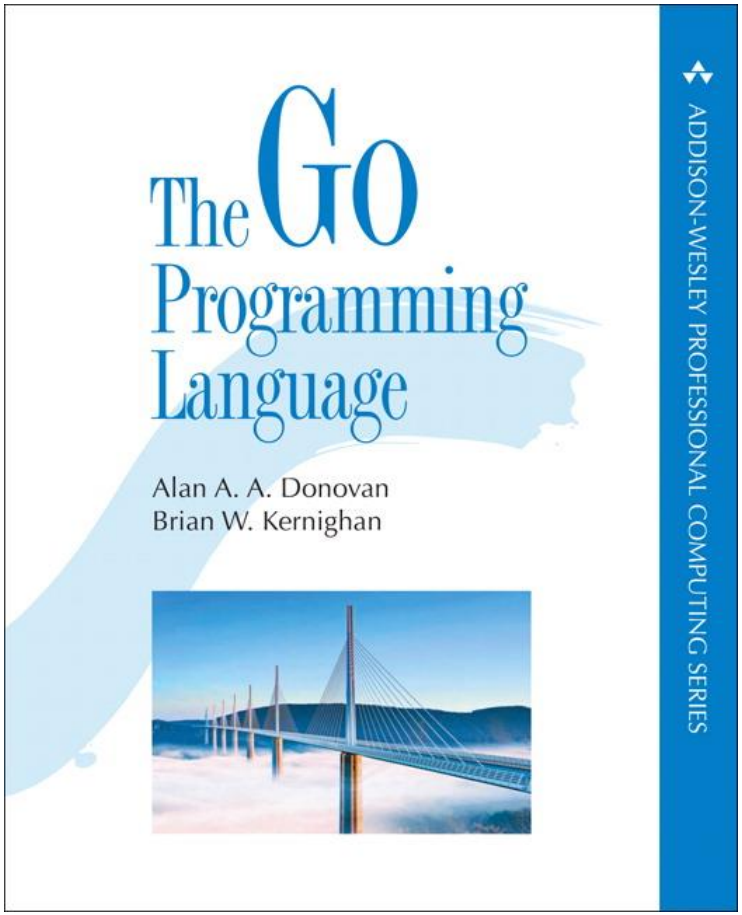
Ph.D. Assistant
- Nikolaus Manes Funk, Dipl. Ing, BSc
- Stephan Felber, Dipl. Ing, BSc

Tutors:
- Markus Paoli
- Konstantin Röhrl, BSc
- Felicia Schmidt
- Richard Steininger, BSc

Technical support:
- Markus Hinkel

Parallel Computing research group, 191-4, www.par.tuwien.ac.at

A book…

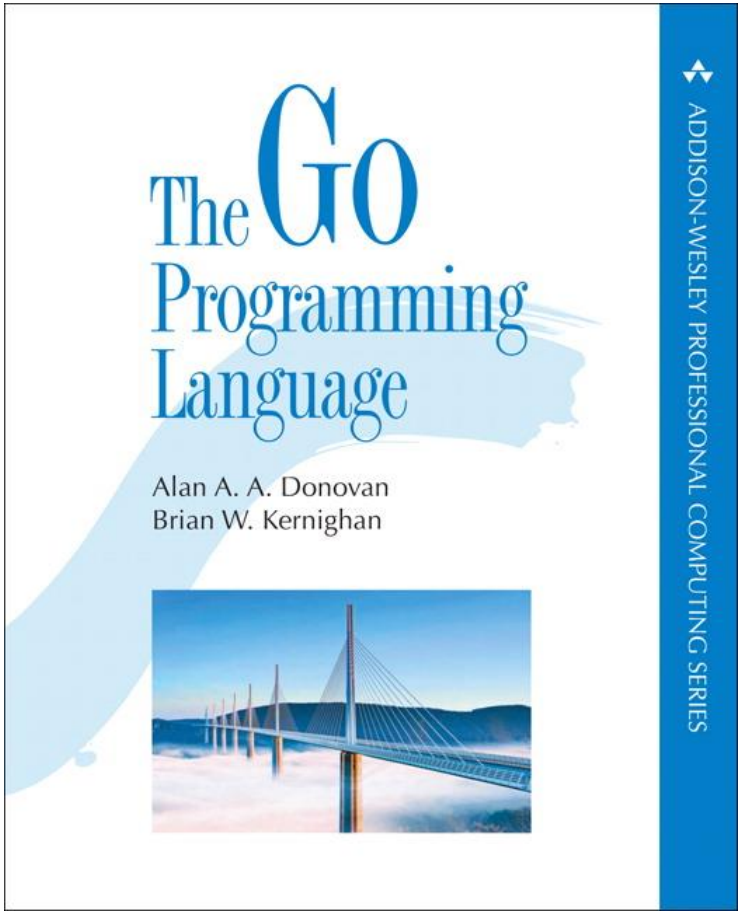

The Go Programming Language
Alan A. A. Donovan
Brian W. Kernighan
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

> Concurrent programming, the expression of a program as a composition of several autonomous activities, has never been more important than it is today. … use concurrency … to exploit a modern computer's many processors, which every year grow in number but not in speed.
>
> … reasoning about concurrent programs is inherently harder than about sequential ones, and intuitions acquired from sequential programming may at times lead us astray.

From Alan A. Donovan, Brian W. Kernighan: The GO Programming Language. Addison-Wesley, 2016

This lecture is not about GO.

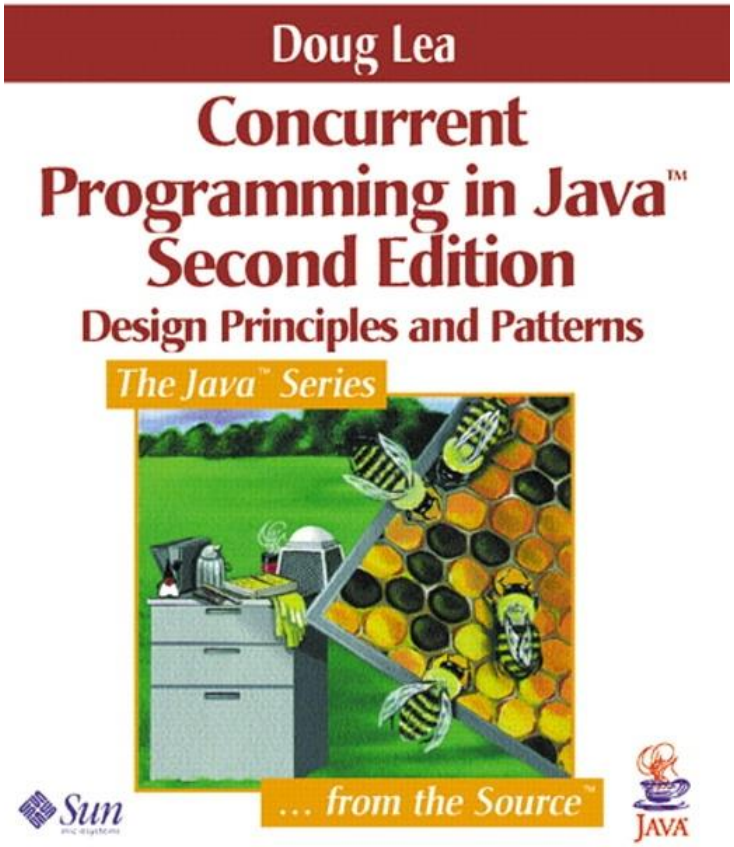
But some of what you will learn will help for certain kinds of programming with GO



GO incorporates some concepts you will learn about in this lecture (message passing, threads, and shared-memory synchronization)

This lecture is also not about Java.

But some of what you will learn will help for certain kinds of programming with Java



AMP SS2022

See also Master Lecture on Advanced Multiprocessor Programming

Concurrent programming ≠ Parallel Computing

~~Concurrent~~ **Parallel** **programming**, the expression of a program as ~~a composition of~~ several ~~autonomous~~ **parallel** **activities**, has never been more important than it is today. … use ~~concurrency~~ **parallelism** … to exploit a modern computer's many processors **efficiently**, which every year grow in number but not in speed.

… reasoning about ~~concurrent~~ **the performance of parallel programs** is inherently harder than about sequential ones, and intuitions acquired from sequential programming may at times lead us astray.

Why parallel computing?

Parallel computers are everywhere, every computer scientist ("Informatiker") must know something about them:

- Why is that?
- What are they good for?
- What exactly is a parallel computer?
- How can we use them efficiently?
- How do we program them?
- What are their limitations?

That's why Parallel Computing is mandatory in the "Software and Information Engineering" Bachelor program (033 534)

… because parallel computing is core computer science

Computer science (my definition): How to use computational resources to solve problems efficiently (in theory and in practice). Parallel computing is computer science with the extra dimension of "parallelism":

- Computer architecture, models
- Algorithms and data structures
- Semantics
- Programming languages, compilers
- Programming, software engineering

A great chance to revisit computer science topics in a new light

Parallel computing: Old discipline, with many challenging, unsolved problems, still lively and highly relevant

Parallel computing at TU Wien (≠ parallel programming)

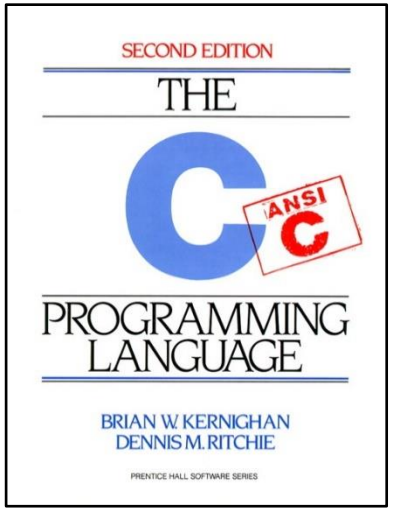
Using parallel computers efficiently

- Aims, motivation, history (“Moore’s law”), basics (time, work, and cost; speed-up; Amdahl‘s Law; scaling), problems and algorithms
- Shared memory parallel computing
- Concrete language: (p)threads, OpenMP (Cilk)
- Distributed memory parallel computing
- Concrete interface: MPI (Message-Passing Interface)
- Newer architectures, new languages (GPU, CUDA, OpenCL)

Prerequisites

Basics on

- Programming, programming languages (we will use C/C++)
- Algorithms and data structures, asymptotic worst-case analysis of algorithms O(f(n)), Ω(f(n)), Θ(f(n))
- Computer architecture (caches, memory)
- Operating systems



Material

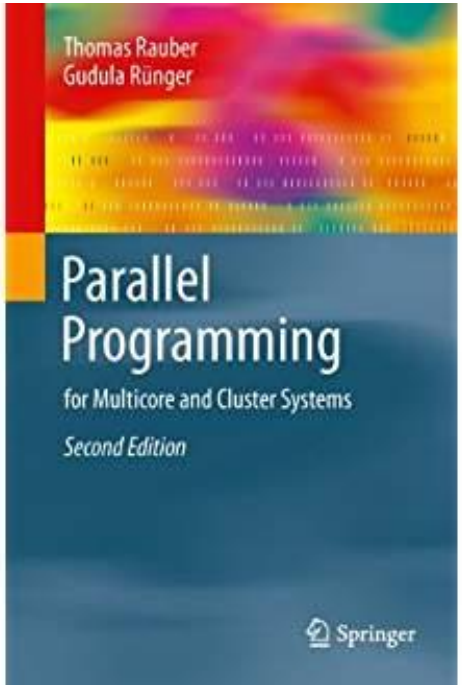
This lecture now comes with a

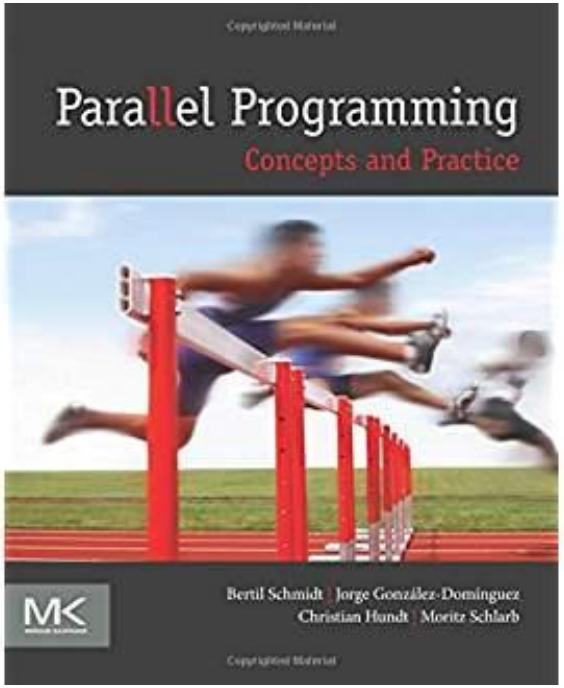
Script
“Lectures on Parallel Computing”

Feedback is welcome!

- Lectures
- Additional online-tutorials
- Slides (many of them)
- Script
- Exercises and programming projects

Helpful books on parallel programming/computing





But: Much which is in the lecture is not in these books and vice versa…