Parallel Computing and I/O

Lecture Parallel Programming

Parallel programming is becoming increasingly important, as even phones and laptops have several processor cores. Some supercomputers even consist of several million cores and have established themselves as a useful and indispensable tool for many areas of science. The resulting analyses and simulations have made it possible to significantly increase scientific insight in many areas.

However, the optimal use of these components is no easy task, which is why scientists are constantly faced with new challenges when developing efficient applications. A deeper understanding of the hardware and software environment as well as the possib causes of errors is therefore essential for parallel programming.

In the lecture, the basics of parallel programming are taught; the exercises serve the practical application and implementation of the acquired knowledge in the C programming language.

The lecture will cover some of the most important topics: Hardware and software concepts (multi-core processors, processes/threads, NUMA etc.), different approaches to parallel programming (OpenMP, POSIX threads, MPI) as well as tools fe performance analysis and debugging (scalability, deadlocks, race conditions etc.). Moreover, reasons and solutions for performance problems are discussed and alternative approaches to parallel programming are presented. Examples and problem are illustrated using real scientific applications.

- ▸ Lecture: Wednesday 17:15-18:45 Uhr (G29-335)
- ▸ Exercises: Wednesday 11:15-12:45 Uhr (G29-335)
- ▸ Contact: Michael Kuhn und Michael Blesel

Participants will learn how to create parallel programs using various programming approaches, how to execute them and how to optimize their execution. In addition, further concepts for parallelization are taught and put into practice in the exercises.

Required skills:

- ▸ Practical knowledge of a programming language and the ability to create simple applications

Recommended skills:

- ▸ Basic knowledge about operating systems
- ▸ Basic knowledge about parallel programming

- ▸ 2024-04-10: Introduction
- ▸ 2024-04-17: Performance Analysis and Optimization
- ▸ 2024-04-24: Hardware Architectures
- ▸ 2024-05-01: Public holiday
- ▸ 2024-05-08: Parallel Programming
- ▸ 2024-05-15: Skipped
- ▸ 2024-05-22: Programming with OpenMP
- ▸ 2024-05-29: Operating System Concepts
- ▸ 2024-06-05: Programming with POSIX Threads
- ▸ 2024-06-12: Programming with MPI
- ▸ 2024-06-19: Networking and Scalability
- ▸ 2024-06-26: Advanced MPI and Debugging
- 2024-07-03: Parallel I/O
- ▸

- 2024-07-10: Research Talks and Debriefing

- 2024-04-10: Introduction
  - Deadline: 2024-04-21, 23:59
- 2024-04-22: Debugging
  - Deadline: 2024-05-05, 23:59
- 2024-04-29: Performance Optimization of a Serial Application
  - Deadline: 2024-05-19, 23:59
- 2024-05-20: Parallelization with OpenMP and Parallelization Schema
  - Deadline: 2024-06-02, 23:59
- 2024-06-03: Parallelization with POSIX Threads
  - Deadline: 2024-06-16, 23:59
- 2024-06-10: MPI Introduction
  - Deadline: 2024-06-23, 23:59
- 2024-06-17: Parallelization with MPI
  - Deadline: 2024-07-07, 23:59

- High Performance Computing: Modern Systems and Practices (Thomas Sterling, Matthew Anderson and Maciej Brodowicz)
- Parallel Programming: for Multicore and Cluster Systems (Thomas Rauber and Gudula Rünger)
- Parallel Programming: Concepts and Practice (Dr. Bertil Schmidt, Dr. Jorge Gonzalez-Dominguez, Christian Hundt and Morit Schlarb)