Course Description High Performance Computing

Keywords: Parallel programming, algorithms and performance

Audience:	**Semester AIM1 and AIM2**	**Module number:**	**AIM 800 6618**
Workload:	**5 ECTS**		**150 h**
divided into	**Contact time**		**50 h**
	Self-study		**60 h**
	Exam preparation		**40 h**
Course language:	**English or German**		
Modul responsibility:	**Prof. Dr.-Ing. Rainer Keller**		
Valid as if:	**14.01.2021**		

Recommended requirements:

Working knowledge programming C / C++

Desired learning outcome of this module:

Students are able to parallelize their own code for CPU and GPU using the various paradigms introduced for shared- and distributed memory systems. They are able to apply the programming paradigms to other code bases and have the practical experience to use the introduced tools.

Knowledge – professional competences

Students know:

- Foundations of parallel and concurrent applications:
  - Communication using fast network interconnects
  - Modern synchronization mechanisms
  - Pitfalls such as deadlocks, priority inversion, etc.
- Tools, Technologies and frameworks for parallel programming,
- Architectural differences of CPUs, GPUs and other accelerators.

Skills – methodical competences

Students are able to:

- Judge on and choose strategies on parallelization techniques for software systems,
- Use various programming interfaces, such as OpenMP, MPI, OpenCL and OpenACC for GPU programming,
- Use parallel debugging tools,
- Do performance analysis on parallel systems.

Comprehensive competences

Students are able to

- Enhance the performance with the help of parallel programming,
- Detect and amend errors in parallel programs,
- Apply the knowledge to other fields and programming languages.

Contents:

- Introduction into parallel programming
- Parallel programming using threads & OpenMP and concurrent Code
- Parallel programming using MPI
- Parallel programming for GPUs using OpenCL and OpenACC
- Efficiency of parallel algorithms
- Performance analysis tools
- Parallel Debugging tools

Literature:

- David R. Butenhof: Programming with POSIX Threads, Addison-Wesley, ISBN 0201633922
- Paul Butcher: Seven Concurrency Models in Seven Weeks, The Pragmatic Programmer, ISBN 1937785653
- Georg Hager, Gerhard Wellein: Introduction to High Performance Computing for Scientists and Engineers. Chapman&Hall, ISBN 978-1439811924
- OpenMP specification
- MPI-Standard
- OpenCL Standard

Offered:

Every summer semester

Submodules and Assessment:

Type of instruction:	Lecture with exercises
Type of assessment:	Exam (90 minutes)
Hours per week:	4 SWS
Estimated student workload:	150 hours

Generation of the module grade:

Exam