



Universität Ulm

Master of Science Finance (PO 2017)

High Performance Computing I

Code 8820572367

ECTS-Punkte 8

Präsenzzeit 6

Unterrichtssprache deutsch/englisch

Dauer 1 Semester

Turnus jedes Wintersemester

Modulkoordinator Prof. Dr. Karsten Urban (Universität Ulm)

Dozent(en) Dr. Andreas Borchert, Dr. Michael Lehn (beide Universität Ulm)

Einordnung in die Studiengänge Computational Science and Engineering M. Sc., 1. - 2. Semester, Pflicht
Mathematik M.Sc., Wahlpflicht Angewandte Mathematik (Numerik)
Mathematische Biometrie M.Sc., Wahlpflicht Mathematik (Numerik)
Wirtschaftsmathematik M.Sc., Wahlpflicht Numerik
Physik M.Sc., Wahlpflicht Mathematik
Wirtschaftsphysik M.Sc., Wahlpflicht Mathematik

Vorkenntnisse Grundmodule in Höherer Mathematik, Numerik und Programmieren

Lernziele Die Studierenden sollen

- den Aufbau und den Betrieb eines Parallelrechners kennen
- die numerischen Softwaretechniken auf Parallelrechnern sicher beherrschen
- die Parallelisierung numerischer Algorithmen beherrschen
- bekannte Verfahren in effiziente Software umsetzen können

Inhalt

- Parallele Rechnerstrukturen
- Formale Speichermodelle mit Anwendungen
- Potentiale und Auswirkungen optimierender Übersetzer
- Techniken zur Parallelisierung numerischer Methoden

Literatur

- H. Schwandt: Parallele Numerik, Vieweg 2003
- M. Herlihy: The Art of Multiprocessor Programming, Morgan Kaufmann, 2012
- M.J. Quinn: Parallel Programming in C with MPI and OpenMP, McGraw- Hill, 2008
- A. Buttari, J. Dongarra, J. Kurzak, J. Langou, P. Luszczek, S. Tomov, The Impact of Multicore on Math Software, <http://icl.cs.utk.edu/projectsfiles/sans/pubs/para-multicore-2006.pdf>
- T.M. Smith, R.A. van de Geijn, M. Smelyanskiy, J.R. Hammond, F.G. van Zee, Anatomy of High-Performance Many-Threaded Matrix Multiplication, http://www.cs.utexas.edu/users/flame/pubs/blis3_ipdps14.pdf
- R.A. van de Geijn, F.G. van Zee, G. Quintana-Orti, Restructuring the Tridiagonal and Bidiagonal QR Algorithms for Performance, <http://www.cs.utexas.edu/users/flame/pubs/RestructuredQRTOMS.pdf>

Lehr- und Lernformen

Vorlesung (2 SWS), Übung (2 SWS), Programmierübungen (2 SWS)

Arbeitsaufwand

28 h Vorlesung (Anwesenheit)
56 h Übungen (Anwesenheit)
156 h Selbststudium
Summe: 240 h

Bewertungsmethode

Die Vergabe der Leistungspunkte erfolgt aufgrund des Bestehens einer mündlichen Prüfung. Die Anmeldung zu dieser Prüfung setzt einen Leistungsnachweis voraus. Als Zulassungsvoraussetzung zur Prüfung (Vorleistung) gilt: Es müssen 50% der Punkte in den Übungsaufgaben (evtl. mit Vorrechnen) erreicht werden.

Notenbildung

Die Modulnote fließt gewichtet mit den ECTS-Punkten in die Gesamtnote ein.

Grundlage für

High Performance Computing II, Vertiefung in CSE
