

Numerische Mathematik 1

Wintersemester 2014/15

Organisation

Lehrveranstaltungs-

leiter: Für die Vorlesung

a.o.Univ.Prof. Mag.Dr. Stephen Keeling, <http://math.uni-graz.at/keeling/>
und für die Übungsgruppen,

a.o.Univ.Prof. Mag.Dr. Stephen Keeling, <http://math.uni-graz.at/keeling/>
Ass.Prof. Dr. Laurent Pfeiffer, <http://www.uni-graz.at/~pfeiffel/>

Unterlagen: Die aktuellsten Unterlagen erscheinen auf der Webseite

<http://math.uni-graz.at/keeling/teaching.html>.

Termine: Der aktuellste Plan der Termine für die Vorlesung und für die Übungen erscheint im KFUOnline.

Vorlesung

Skriptum: Die Vorlesung orientiert sich am folgenden Skriptum:

http://math.uni-graz.at/keeling/num1_ws14/numerik.pdf

Note: Es werden Termine für eine schriftliche Prüfung nach dem Ende des Semesters angeboten.

Übungen

Aufgabenblätter: Die aktuellsten Aufgabenblätter erscheinen auf der Webseite

<http://math.uni-graz.at/keeling/teaching.html>.

Die Aufgaben sollen bis zum angegebenen Termin bearbeitet und die Lösungen in der Übungseinheit vorgetragen werden.

Programmier-

aufgaben: Die Teilnehmer der Gruppen *Keeling* und *Pfeiffer* sollen ihre Lösungen der Programmieraufgaben per Email an stephen.keeling@uni-graz.at beziehungsweise laurent.pfeiffer@uni-graz.at schicken. Es werden nur getestete und lauffähige Lösungen akzeptiert.

Werkzeug: Zur Bearbeitung empfiehlt sich besonders Matlab, das auf den Rechnern im Computerraum installiert ist. Sie können sich eine kurze Einführung in Matlab unter

http://www.uni-graz.at/keeling/num1_ws14/MatlabPrimer.pdf

herunterladen. Für guten Programmierstil in Matlab berücksichtigen Sie auch die Hinweise auf

http://www.uni-graz.at/keeling/num1_ws14/matlab_style_guidelines.pdf.

Möchten Sie auch auf Ihrem eigenen Rechner an den Aufgaben arbeiten, und besitzen Sie keine Studentenversion von Matlab, so können Sie das kostenlose und quelloffene GNU Octave verwenden, das unter

<http://www.gnu.org/software/octave/>

verfügbar ist. Es ist (für die Zwecke dieser Vorlesung) nahezu vollständig kompatibel zu Matlab.

Punktesystem: Zu Beginn jeder Übungseinheit kreuzt man auf einer Liste der Hausaufgaben die Beispiele an, die man bereit ist, an der Tafel zu lösen. Jedes Kreuz zählt 1 Punkt. Die Tafelleistung wird mit bis auf 4 Punkten beurteilt. (Wohl Durchgedacht? Eigener Beitrag? Diskussion?) Zusätzlich bekommt man 1 Punkt für jeden konstruktiven Beitrag während der Übungseinheit. Eine Programmieraufgabe wird mit bis auf 4 Punkten beurteilt. (Funktioniert? Löst Test-Beispiele? Kommentiert? Leicht zu lesen?)

Note: Für eine positive Note muss man mindestens einmal an der Tafel aufgetreten haben, und alle Programmieraufgaben müssen gelöst worden sein. Sonst basieren die Noten auf die Verteilung der gesamten Anzahl von Punkten von Teilnehmern.