

Matlab-Blatt 3

(Abgabe bis spätestens Dienstag, den 06.12.2011, 10:00 Uhr per Mail (s.u.).)

Aufgabe 1 (*Instabilitäten*)

(17+6 Punkte)

Zur Approximation der Exponentialfunktion kann die wohlbekannte Reihendarstellung

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

verwendet werden.

- a) Schreiben Sie eine Funktion `e_approx = myExpApprox(x)`, welche diese Approximation implementiert. Ist es dabei sinnvoll, $k!$ direkt zu berechnen? Kommentieren Sie dies und überlegen Sie sich eine alternative Methode für die Berechnung der Zuwächse $\frac{x^k}{k!}$, welche die vollständige Neuberechnung der Summanden in jedem Schritt umgeht.

Hinweis:

- Die Reihe kann sinnvollerweise abgebrochen werden, wenn die Zuwächse kleiner als die Maschinengenauigkeit sind (vgl. das Abbruchkriterium in Funktion `mineps.m` auf Seite 38 im Skript).

- b) Schreiben Sie ein Skript `main1.m`, mit dem Sie die obige Berechnungsmethode mit der direkten Auswertung von e^x für verschiedene $x \in [-30, 30]$ vergleichen können. Plotten Sie die absoluten und relativen Fehler. Was beobachten Sie? Was für eine Erklärung könnte es dafür geben?

Hinweis:

- Die Fehler lassen sich am sinnvollsten in einem Plot mit logarithmischer y -Achse darstellen. Der entsprechende Matlab-Befehl dazu lautet `semilogy`.

- c) Verwenden Sie die Beobachtungen aus (b) und modifizieren Sie Ihre Funktion aus a) so, dass Sie für alle $x \in \mathbb{R}$ einen stabilen Algorithmus bekommen. Nennen Sie diese neue Funktion `myBetterExpApprox(x)`.

Hinweis: Fallunterscheidung

- *d) **Bonusaufgabe:** Modifizieren Sie beide Funktionen so, dass Sie als Eingabe `x` einen Vektor verwenden können. Nennen Sie diese Funktionen `myExpApproxVecVersion(x)` bzw. `myBetterExpApproxVecVersion(x)`

Aufgabe 2 (*Factorial Number System*)

(5 Punkte)

In exercise 3(a) on theory sheet T3, you formulated an algorithm to add two numbers in the factorial based representation. Here, we will only consider (positive) integers, i.e. numbers $x = (a_r \dots a_1)_F$ with

$$x = \sum_{i=1}^r a_i \cdot i!$$

Implement your algorithm in a function `z = addFactorialNumbers(x,y)` that takes as arguments two vectors `x` and `y` containing the coefficients $a_{r_x}^x, \dots, a_1^x$ and $a_{r_y}^y, \dots, a_1^y$, respectively, and returns the vector `z` with the coefficients of the sum of x and y .

Test your algorithm on `x = [2 2 0]`, `y = [2 1 1]`. (The result should be `z = [1 1 0 1]`).

Senden Sie alle Dateien (*m-Files, Plots, Erklärungen*) in einer Email mit dem Betreff `Num1-BlattM3` an `kristina.steih@uni-ulm.de`. Bitte alle Dateien in ein zip-File packen, welches die Namen der Studenten enthält, also z.B. `M3_Student1_Student2.zip`. Aus der Email sollte zusätzlich klar hervorgehen, von welchen beiden Studenten die Lösung ist. Bitte schreiben Sie außerdem dazu, in welcher Übungsgruppe (Wochentag und A bzw. B) Sie jeweils sind.