KV Wissenschaftliches Rechnen, WS 2024/25

Hausübung 1

Die Aufgaben können in C oder C++ gelöst werden, wobei neuere C++ Standards/Standardbibliotheken nach Belieben verwendet werden dürfen. Abzugeben sind

- ein gemeinsames oder drei unabhängige `.cpp` oder `.c` Files, enthält Quellcode,
- ein `.pdf` File, enthält alle Ausgaben/Antworten in Form eines Protokolls,
- zusätzlich, nicht zwingend erforderlich `CMakeLists.txt`.

Die Aufgaben dürfen in Gruppen von bis zu 3 Personen bearbeitet werden. In diesem Fall gibt einer der Teilnehmer die Lösungen im Moodle-Kurs ab, und die Namen der anderen Studierenden werden im `.pdf` vorne vermerkt.

Aufgabe 1 - 2 Punkte Berechnen Sie den Wert $\log(x)$ für verschiedene x über die Summendarstellungen

- Taylorreihe um $x_0 = 1$, Konvergenzbereich $0 < x \leq 2$

$$\sum_{n=1}^{N} \frac{(-1)^{n+1}(x-1)^n}{n}$$

- Reihe, Konvergenzbereich $x > 0.5$

$$\sum_{n=1}^{N} \frac{((x-1)/x)^n}{n}$$

- Reihe, Konvergenzbereich $x > 0$.

$$2\sum_{n=1}^{N} \frac{((x-1)/(x+1))^{2n-1}}{2n-1}$$

Testen Sie für $x = 1.1$, $x = 0.51$, $x = 0.1$, $x = 1.9$. Wie groß muss N jeweils gewählt werden, damit der absolute Fehler kleiner 10^{-6} ist (falls konvergent; vergleichen Sie dafür mit dem Wert der Standard-Log-Funktion).

Aufgabe 2 - 2 Punkte Implementieren Sie eine Funktion, die bestimmt, ob drei gegebene Vektoren $\vec{a}, \vec{b}, \vec{c} \in \mathbb{R}^3$ linear abhängig sind,
`bool LinearDependent(const double* a, const double* b, const double* c)`.
Überlegen Sie ein geeignetes Toleranzkriterium. Es dürfen alle Vektoren statisch mit Länge 3 allokiert werden, Speicherallokation passiert ausschließlich außerhalb der Funktion.

Testen Sie die Funktion für

- unabhängig: $\vec{a} = [1, 2, 3]$, $\vec{b} = [-0.6, 3., 5]$, $\vec{c} = [1, 2.2, -2.4]$
- abhängig: $\vec{a} = [1/3, 2/3, 1]$, $\vec{b} = [-10^8, 5 \cdot 10^8, 0]$, $\vec{c} = [0, 1, 3/7]$
- unabhängig: $\vec{a} = [1/3, 2/3, 1]$, $\vec{b} = [-10^{-8}, 7 \cdot 10^{-8}, 0]$, $\vec{c} = [0, 1, 3/7]$

Aufgabe 3 - 2 Punkte Lesen Sie eine Liste von Gleitkommazahlen aus der Datei `data_win.txt` (auf Windows; `data_unix.txt` auf Linux oder Macos) ein, wobei Sie davon ausgehen können, dass die Zahlen zeilenweise enthalten sind und keine weiteren (Trenn-)Zeichen in der Datei enthalten sind. Die Länge der Liste ist dabei unbekannt (d.h. das Programm sollte auch funktionieren wenn Zahlen dazugefügt oder gelöscht werden). Erstellen Sie eine weitere Datei `output.txt`, in die Sie diese Zahlen in einer einzigen Zeile, durch Komma `','` getrennt, in wissenschaftlicher Darstellung mit 6 Nachkommastellen schreiben (also etwa `1.239400e+00`). Nach der letzten Zahl darf noch ein Komma angefügt werden. Diese Aufgabe kann direkt in der `main` Routine gelöst werden, es darf aber auch eine entsprechende Funktion implementiert werden.