

Übungen zur Computerphysik

SS 2020

T. Luu, A. Nogga, M. Petschlies, A. Wirzba

Übungsblatt 1

9. - 15. April 2020

A.1: Fingerübungen zur C-Programmierung

1. I/O und Bearbeiten von Daten

Zum Aufwärmem wollen wir etwas C-Code schreiben mit einigen grundlegenden Funktionen, die für die Bearbeitung späterer komplexerer Aufgaben sehr nützlich sind: Daten (in Text-/Binärformat) einlesen, im Speicher halten, ausgeben, und (selbst geschriebene) Funktionen darauf anwenden.

Referenzen für Funktionen gibt es z.B. auf https://en.wikibooks.org/wiki/C_Programming/Standard_library_reference (oder einfach per Google-Suche).

- (a) Unter dem Link dataset1 finden Sie eine Datei mit einem Datensatz (1. Zeile Integer gibt die Länge des darauf folgenden Datensatzes). Schreiben Sie ein C-Programm, dass den Datensatz einlesen und auf dem Bildschirm ausgeben kann. (Speichern Sie die Daten in einem Array ab.)

Vgl. Beispielprogramm A1.c, Funktionen `read_dataset` (`if-branch binary == 0`), `show_dataset`.

Die Compiler-Optionen (wie `-Wall`, `-W...`, `-pedantic` , ...) können z.B. unter <https://gcc.gnu.org/onlinedocs/gcc/Invoking-GCC.html> nachgelesen werden.

- (b) Implementieren Sie die Funktion “kumulative Quadratsumme”

$$\sigma_k = \sum_{i=0}^k A_i^2, \quad k = 0, \dots, N-1, \quad (1)$$

und wenden Sie sie auf den eingelesenen Datensatz $\{A_i\}$ der Länge N an. Speichern Sie das Resultat in einer separaten Datei als Datensatz ab. (Hinweis: C-Funktionen `fprintf`, `fscanf`).

Vgl. Beispiel-Funktion `cum_sqr_sum` in A1.c

- (c) Erweitern Sie Ihr Programm zum Einlesen des binären Datensatzes dataset2 (1 Integer mit der Länge des Datensatzes, darauf folgend der Datensatz selbst in double precision). Speichern Sie in Tabellenform (A_i , $\sin(A_i)$) in einer Datei ab. (Hinweis: C-Funktion `fread`; Zur Erinnerung: Aufruf von `sin(...)` benötigt das Einbinden der Mathe-Bibliothek über Header-Datei `math.h` und Linker-Befehlszusatz `-lm`.)

Vgl. Funktion `read_dataset` (`if-branch binary == 1`)

2. Plotten von Daten

Die grafische Darstellung von Daten und Präsentation sind ein wichtiges Hilfsmittel für deren Interpretation. Machen wir auch dazu ein Beispiel.

- (a) Erstellen Sie eine Datei mit der grafischen Darstellung der in (c) gespeicherten Tabelle (z.B. mit gnuplot). Achten Sie dabei auf die Achsenbeschriftung und die Erklärung der verwendeten Plot-Symbole.

Vgl. Beispiel gnuplot Plotskript plot.gpl

- (b) Erstellen Sie ein kleines PDF-Dokument (z.B. mit latex), das den in (a) erhaltenen Plot zeigt sowie eine kurze Beschreibung desselben.

Vgl. Beispiel latex-Datei A1.tex

3. *Optional:* Versionskontrolle mit git

“Version control” ist das A und O für die organisierte Erstellung und Verwaltung von Computercode (aber auch Dokumenten und vielen anderen Projekten). Das Programm git bietet dazu eine Plattform.

- (a) Gewinnen Sie erste Einblicke in die Funktionsweise von und die Arbeitsweise mit git, z.B. auf <https://www.atlassian.com/de/git>.

- (b) Erstellen Sie sich ein Repository und pflegen Sie Ihren erstellten Code als erstes Projekt ein.

(Hinweis: im wesentlichen die Befehle git init , git add , git commit; git-Spickzettel z.B. unter

<https://www.atlassian.com/git/tutorials/atlassian-git-cheatsheet>)

Bei weitergehendem Interesse geben wir gern zusätzliche eine Übersicht. Ansonsten bitte einfach mal ausprobieren.