

Dokumentation Programm test

Armin Ulrich

14. November 2017

1 Sinn des Programms

Dieses Programm ist als Einstieg gedacht: der Benutzer soll den Umgang mit den Programmen und der Octave-Umgebung bzw. das Compilieren mit einem Makefile kennenlernen. Das Programm ist nicht weiter tiefsinnig: es liest zwei Fließkommazahlen von der Tastatur ein und gibt die Summe aus beiden auf dem Bildschirm aus.

2 Anwendung des Programms

Um die hier gegebenen Programme laufen zu lassen, sollten Sie wie folgt vorgehen:

1. Zuerst entscheiden Sie sich für eine der Programmiersprachen Octave, python3, fortran 95 oder C. Sie können natürlich auch alle vier Sprachen ausprobieren, wenn Sie wollen.
2. Dann erstellen Sie ein Verzeichnis, in dem Sie dieses Testprogramm laufen lassen wollen.
3. In diesem Verzeichnis benötigen Sie die Quellcode-Dateien, d.h. die Octave-Datei *test.m*, die fortran 95-Datei *test.f95*, die C-Datei *test.c* oder die python3-Datei *test.py*. Da fortran 95- und C-Dateien compiliert werden müssen, benötigen Sie dann auch die dazugehörigen Makefiles. Sie können die Dateien entweder einzeln anklicken und speichern oder das Archiv herunterladen und im Verzeichnis öffnen. Das Archiv enthält den Makefile und die jeweilige Quelldatei und die Dokumentation als *.pdf. Der Makefile beinhaltet die Befehle zum Erstellen der ausführbaren Datei und ist spezifisch für die jeweilige Programmiersprache. Eine C-Quelldatei kann nicht mit dem fortran 95-Makefile compiliert werden und umgekehrt. Im C-Verzeichnis liegt der richtige Makefile für den C-Quellcode bereit, ebenso im fortran 95-Verzeichnis.
4. Dann öffnen Sie ein Konsolenfenster in diesem Verzeichnis (rechter Mausklick im Fenster des Verzeichnisses, dann **Terminal hier öffnen** klicken).
5. Führen Sie einen Virensan über dieses Verzeichnis durch. Z.B. ich benutze clamscan mit dem Befehl **clamscan -r -i**. Sie können natrlich einen anderen Virens Scanner benutzen.

6. Das Vorgehen bei Benutzung eines Quellcodes in fortran 95 und C unterscheidet sich von dem bei Benutzung eines Octave-Codes bzw. eines python3-Codes:
 - (a) Die fortran 95- oder C- Quelldatei wird mit `make` kompiliert, es wird eine ausführbare Datei erzeugt.
 - (b) Führen Sie einen erneuten Virenskan über dieses Verzeichnis durch.
 - (c) Mit `./test` startet man das Programm in der Konsole.
7. Vorgehen bei Octave:
 - (a) In der Konsole wird `octave` aufgerufen.
 - (b) Im Befehlsfenster wird mit `test` das Programm gestartet.
8. Vorgehen bei python3:
 - (a) In der Konsole wird das Programm mit `python3 test.py` gestartet.
9. Das Programm bittet Sie, nacheinander zwei Fließkommazahlen einzugeben. Die Summe aus beiden wird berechnet und angezeigt.

3 Physikalischer Hintergrund

Entfällt.

4 Mathematischer Hintergrund

Der ist natürlich sehr kurz: $z = x + y$

5 Ergebnis des Programms

Auch hier gibt es nicht viel zu dokumentieren. Das Programm gibt die Summe der beiden Zahlen auf dem Schirm aus, mit begrenzter Stellenzahl. Es wird hin und wieder vorkommen, daß statt einer glatten Zahl eine Zahl mit einer Menge Neunen hinter dem Dezimalpunkt ausgegeben wird. Mit numerischen Fehlern werden wir uns im weiteren Vorgehen beschäftigen.

6 Fehlerbetrachtung

Aus der begrenzten Stellenzahl folgt, da unter Umständen Dezimalstellen einer Zahl unterdrückt werden.

7 Struktogramm

Variablendeklaration: Fließkomma: x,y,z
A(B): 'Geben Sie bitte eine Zahl x (Dezimalpunkt zulässig) ein:'
E(T): x
A(B): 'Sie haben die Zahl', x , 'eingegeben.'
A(B): 'Geben Sie bitte eine weitere Zahl y (Dezimalpunkt zulässig) ein:'
E(T): y
A(B): 'Sie haben die Zahl', y , 'eingegeben.'
$z = x + y$
A(B): 'Die Summe der Zahlen x und y ist:', z
Programmende