

Programmieren für Physiker

Interfakultatives Institut für Anwendungen der Informatik
Institut für Theoretische Teilchenphysik

Prof. Dr. M. Steinhauser, Dr. A. Mildenerger
<http://comp.physik.kit.edu/>

SS 2017 – Blatt 01
Bearbeitungszeitraum: bis 03. Mai 2017

Die richtige Bearbeitung der Pflichtaufgaben wird von den Tutoren während der Übungszeit testiert. Zulassungsvoraussetzung zur Klausur gegen Ende des Semesters ist die richtige Bearbeitung von 80% der Pflichtaufgaben.

Aufgabe 1: Kennenlernen des Systems

freiwillig

Im Poolraum Physik ist das Betriebssystem Linux mit der grafischen Benutzeroberfläche KDE installiert. Neben dem Arbeiten mit der Maus lassen sich viele Dinge auch per Textbefehle erledigen, häufig sogar schneller, falls man die benötigten Befehle gut kennt. In dieser Aufgabe finden Sie einige oft gebrauchte Kommandos.

Um Befehle eingeben zu können, benötigen Sie ein sogenanntes Terminal (Textfenster). Dieses erhalten Sie in der grafischen Benutzeroberfläche im Menü unten links im ersten Reiter oder durch Navigieren Applications→System→Terminal.

- Hilfe zu den meisten Befehlen in der Kommandozeile erhalten Sie mit dem Befehl `man`. Als Beispiel hierfür können Sie sich die „Hilfe zur Hilfe“ anschauen: `man man`. `man` wird oft benötigt, um Optionen von Befehlen nachzuschlagen. Verlassen Sie die Hilfeanzeige durch Drücken von `q`.
- In Ihrem „Heimat-Verzeichnis“ bzw. „Home-Directory“ (bei neuen Konten in der Form: `/home/ck17/IhrLoginName`) befinden sich einige wenige Verzeichnisse. Benutzen Sie die folgenden Kommandos, um sich ein wenig umzusehen: `ls`, `pwd`, `cd`. Mit vorgestelltem `man` erfahren Sie dabei mehr über die einzelnen Kommandos.
- Das Zeichen `~` dient als Abkürzung für Ihr Home-Directory. Schauen Sie sich an, in welchen Verzeichnissen Sie bei `cd /`; `pwd` und bei anschließendem `cd ~`; `pwd` sind. Der Verzeichnisname `..` steht für das übergeordnete Verzeichnis. Probieren Sie `cd ..`; `pwd`.
- (Text-)Dateien können Sie recht bequem mit dem Befehl `less` anzeigen; Alternativen hierzu sind `more` und `cat`. Laden Sie von der Homepage des Programmierkurses eine C++-Datei herunter und schauen Sie sie mittels `less` an. Sie verlassen das Programm mit der Taste `q`.
- Mit `ls -l` erhalten Sie eine Reihe von Informationen über Dateien im aktuellen Verzeichnis. Was bedeuten die einzelnen Angaben? Wie sind z.B. die Dateirechte am linken Rand der Ausgabe zu lesen? Dateirechte ändern können Sie übrigens mit dem Befehl `chmod`.
- Befehle um Dateien bzw. Verzeichnisse zu kopieren, verschieben, löschen sind: `cp`, `mv`, `rm`, `mkdir`, `rmdir`. Versuchen Sie beispielsweise ein Verzeichnis anzulegen und zu löschen.
- Ihr Passwort können Sie mit dem Befehl `passwd` ändern. Verwenden Sie ein sicheres Passwort: Es sollte mindestens 8 Zeichen umfassen und **nicht** auf einer Zeichenfolge beruhen, die in einem Wörterbuch zu finden ist!
- Versuchen Sie, mit Hilfe des Programmes `ssh` sich von Ihrem Rechner aus in den Rechner Ihres Nachbarn einzuloggen. Vergleichen Sie z.B. den Inhalt der Verzeichnisse `/tmp` auf den beiden Rechnern. Sind die Verzeichnisse identisch?
Mit `scp` können Sie Dateien auf einen anderen Rechner übertragen.
- Es gibt zahlreiche weitere Befehle, die das Arbeiten in Textfenstern schnell und bequem machen, hier eine Auswahl:
`top`, `du`, `df`, `grep`, `locate`, `find`, `apropos`, `whereis`, `info`, `who`, `wc`, `head`, `tail`, `date`, `tar`, `gzip`, `gunzip`, `bzip2`, `bunzip2`, `echo`

Aufgabe 2: Editieren

freiwillig

Kopieren Sie die Datei *editiere.mich* mit dem Befehl:

```
scp fphct01:/tmp/editiere.mich ~
```

in Ihr Home-Directory. Auf dem System installierte Editoren sind unter anderem `kate`, `emacs`, `xemacs`, `joe`, `kwrite`, `pico`, `nano`, `gvim`. Öffnen Sie nun die Datei mit einem Editor Ihrer Wahl und führen Sie die darin enthaltenen Editieranweisungen durch. Versuchen Sie dabei, möglichst viele abkürzende Tastenkombinationen des Editors („Short-Cuts“) zu benutzen.

Für `emacs` und `xemacs` gibt es eine Kurzanleitung, die wichtige Tastenkombinationen auflistet:

```
okular /usr/share/emacs/24.3/etc/refcards/refcard.pdf
gv /usr/share/xemacs/21.5.34/etc/refcard.ps.gz
```

Aufgabe 3: Exponentialfunktion**Pflichtaufgabe**

Schreiben Sie ein Programm `exponent.cc`, welches e^x berechnet und ausgibt. Dabei soll die Variable x vom Benutzer im Programm abgefragt werden und vom Typ `float` sein. Wenn Sie im Programmkopf `#include <cmath>` verwenden, können Sie die Exponentialfunktion mit dem Ausdruck `exp(x)` berechnen.

Compilieren Sie das Programm durch Aufruf von `g++ exponent.cc -o exponent`. Die Option `-o...` bewirkt dabei, dass Ihr ausführbares Programm `exponent` heißt. Starten Sie das Programm durch `./exponent`.

Testen Sie das Programm mit kleinen und großen Werten x um herauszufinden, innerhalb welchen Zahlenbereichs das Programm funktioniert.

Hinweis: Mit dem Rechnernamen `fphctssh.physik.uni-karlsruhe.de` können Sie von überall aus mittels `ssh/scp` Programm per Netzwerk auf einen Poolrechner zugreifen.
