
MATHEMATIK I FÜR BIOLOGEN, GEOLOGEN UND GEOÖKOLOGEN
Computer-Übungsblatt 1

Zur Bearbeitung der Computer-Übungsaufgaben stehen die Pools der Informatik in den Räumen **S1** und **H13** im C-Gebäude 2. Stock (Öffnungszeiten 7:00-19:00 Uhr) zur Verfügung. Sie können die Computer-Aufgaben allerdings auch zu Hause bearbeiten. Die für den Informatik-Pool benötigten Logins und Passwörter erhalten Sie in der Übungsstunde. Bei Fragen zu den Programmier-Aufgaben wenden Sie sich bitte an Michael Fromm,

`michael.fromm@student.uni-tuebingen.de`, Sprechstunde Mo, Mi, Fr 13-15 Uhr im **C2S1**.

Nach dem Einloggen können Sie einen Browser öffnen, indem Sie unten links auf das Netscape-Symbol klicken. Eine erste Hilfe zu den Free-BSD-Rechnern des Pools finden Sie unter <http://www-doc.informatik.uni-tuebingen.de/intro/>. Als Programmiersprache benutzen wir die numerische Skript-Sprache *octave* (<http://www.octave.org/>). Eine erste Einführung kann unter <http://na.uni-tuebingen.de/~lorenz/Octave.pdf> heruntergeladen werden. Wenn Sie diese Datei gespeichert haben, können Sie sie lesen, indem Sie eine Konsole öffnen (klicken auf das Konsolenfenster-Symbol links unten) und `acoread` eingeben. Bitte lesen Sie das erste Kapitel davon, das die Grundlagen der Bedienung von *octave* enthält. Eine weitere, detaillierte Dokumentation findet sich unter <http://www.octave.org/doc/index.html> (auf Englisch). *octave* selbst ist unter <http://sourceforge.net/projects/octave> frei erhältlich. Auf den Info-Rechnern läuft *octave* in der Version 2.1.71 und kann von der Konsole durch die Eingabe `octave` gestartet werden. Sie können nun auf der Kommandozeile von *octave* elementare Befehle (siehe Doku) eingeben oder einfach ein Skript starten. Als erstes aber ein kleines

Beispiel 0:

```
>> a = 2^10 Dies definiert eine Variable a und weist ihr den Wert 210 zu.
```

Beispiel 1:

```
>> x = 0:.1:10; Dies definiert einen Datenvektor x mit 101 Komponenten (x1, x2, ..., xi, ..., x101), mit den Werten (0, 0.1, 0.2, ..., 10), also im Intervall von 0 bis 10, in Schritten von 0.1 .  
>> fx = x.^2; Erzeugt einen Datenvektor fx mit Einträgen fxi = xi2. Ein Punkt vor '^' bewirkt also komponentenweise Operationen auf Datenvektoren, während z.B. für Skalare (z.B. y=5^2) kein Punkt notwendig ist.  
>> plot(x,fx) Zeichnet die Werte in fx als Funktionwerte der xi in x.
```

Aufgabe C1.

Plotten Sie nun die Gauß-Funktion

$$\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (1)$$

im Intervall $[-4, 6]$ für $\mu = 1$ und $\sigma = 2$ anhand des Beispiels 1. Definieren Sie sich hierzu zuerst Variablen `mu` und `sigma`.

Tip: Die Quadratwurzel von `p` errechnet man mit `sqrt(p)`. Wie erhält man jedoch π ?

Aufgabe C2.

Öffnen Sie einen Text-Editor (im Info-Pool öffnen Sie dazu eine neue Konsole und geben `pico` ein). Mit Hilfe des Text-Editors können Sie externe *octave*-Funktionen und Skripte schreiben und diese im *octave*-Verzeichnis abspeichern. Schreiben Sie nun anhand des Beispiels 2 eine Funktion `gauss(x,mu,sigma)`, welche als Eingaben `x` (Datenvektor!), `mu` (Skalar) und `sigma` (Skalar) erhält und den entsprechenden Funktionswert von (1) an der Stelle x ausgibt. Der Aufruf erfolgt unter der *octave*-Konsole durch

```
>> x = -4:.1:6; Unser bekannter Datenvektor...
>> mu = ... , sigma = ... Dies definiert Variablen mu und sigma und weist ihnen Werte zu.
>> fx = gauss(x,mu,sigma) Aufruf der Funktion gauss, deren Ausgabe ein Datenvektor mit
Funktionswerten von (1) ist.
>> plot(x,fx) Zeichnet die  $fx_i$  und  $x_i$  in ein Diagramm
```

Beispiel 2:

Einfache Funktionen und Skripte speichert man unter dem Funktionsnamen ab, in diesem Beispiel also unter `pol.m`

```
function fx = pol(x,a,b,c)
    fx = a*x.^2+ b*x + c;
endfunction
```

`function fx = pol(x,a,b,c)` definiert eine Funktion `pol(x,a,b,c)` mit Eingabewerten `x`, `a`, `b`, `c` und Ausgabewert `fx`; gleichzeitig markiert es den Beginn des Funktions-Körpers.
`fx = a*x.^2+ b*x + c;` definiert den Ausgabewert `fx` und weist ihm das Ergebnis der rechten Seite zu (man beachte den Punkt für komponentenweises Arbeiten).
`endfunction` Ende der Funktion

Abgabe: Einen Ausdruck der von Ihnen mit *octave* erzeugten Resultate fügen Sie Ihren schriftlichen Hausaufgaben bei und geben ihn am **Donnerstag, 3.11.2005** zu Beginn der Vorlesung ab.

Webseite zur Vorlesung: <http://www.maphy.uni-tuebingen.de/lehre/ws-2005-06/m4b>