

Übungen zur Stringverarbeitung: Blatt 3

Programmierung der UNIX-Shell Bash

1. Quotingregeln

Quoting bedeutet das Maskieren eines Strings, so dass er beim normalen Substitutionsprozess der Shell nicht verändert wird (\Rightarrow Vorlesung). Kommt das Quotingzeichen auch im String vor, muss es selbst gequotet werden.

a) Benutzen Sie `awk` von der Shell aus, um folgende Strings auszugeben:

- `"'\`
- `\"'\"`

Der Befehl sieht also in etwa so aus:

```
awk 'BEGIN { print "<<hier wirts interessant>>" }'
```

5 Punkte

b) Eine Shellvariable »winform« enthalte einen Verzeichnisnamen in Windows-Manier. Setzen Sie die Variable »uniform« mit dem Wert von »winform«, wobei alle Backslashes »\« durch Forward Slashes »/« ersetzt sind:

```
winform='c:\windows\system'  
uniform=$( echo "$winform" | sed '<<hier sed Code>>')
```

oder

```
uniform=$( echo "$winform" | awk '{gsub(<<hier awk Code>>)} 1')
```

oder Shell-Substitution. Eine Lösung reicht aus.

5 Punkte

2. Simulation eines Ferrosolitons

Die Datenfiles »soliton00.dat« bis »soliton80.dat« enthalten eine Simulation über das Anwachsen eines Ferrosolitons im Bereich von 0ms bis 80ms. Das Format ist einfach eine zweispaltige Darstellung der Amplitude als Funktion des Abstands vom Zentrum. Sie wollen aus diesen Simulationsdaten möglichst einfach eine Animation erzeugen, wozu Sie Gnuplot verwenden.

a) Schreiben Sie ein Shellskript, das Gnuplot ansteuert um eine Animation mit 0.1s Pause zwischen den Frames anzuzeigen. Die aktuelle Simulationszeit soll dabei im Plotfenster angezeigt werden.

5 Punkte

b) Die Simulationsdaten sind natürlich gefälscht :-). Sie sind einfach aus folgender Funktion generiert:

$$A(x, t) = J_0^A(x) \cdot A(t), \quad (1)$$

wobei J_0 die Besselfunktion ist und $A(t)$ die Lösung folgender Amplitudengleichung:

$$\frac{1}{s} \dot{A} + \ddot{A} = \varepsilon A - gA^3 \quad (2)$$

mit den Parametern

$$\varepsilon = 0.05 \quad g = 0.05 \quad s = 3 \quad (3)$$

$$A(0) = 0.01 \quad \dot{A}(0) = 0 \quad (4)$$

Die Lösung der Amplitudengleichung wurde numerisch mit »ode« bestimmt, das Inputfile liegt in »amplitude.ode«. Der Output ist »zeitdep.dat«. Schreiben Sie ein Shellskript, das mit Hilfe von gnuplot diese Pseudodaten erzeugt.

5 Punkte

Hinweise

- a) Zu 2a: Lassen Sie eine Schleife über alle Datenfiles laufen, die Gnuplot-Befehle ausgibt. Die Befehle sollen dann mit einer Pipe an Gnuplot geschickt werden und sehen in etwa so aus:

```
set yrange [0:1.2]
set xrange [-2*pi:2*pi]
...
plot "soliton20.dat" w l lw 3 title "Zeit 20 ms", 1 notitle
pause 0.1
plot "soliton21.dat" w l lw 3 title "Zeit 21 ms", 1 notitle
...
pause 2
```

Die aktuelle Zeit schneiden Sie aus dem Dateinamen aus.

- b) Zu 2b: Lesen Sie das Datenfile »zeitdep.dat« in einer Schleife zeilenweise ein und generieren Sie plot-Befehle für Gnuplot. Die Daten können Sie statt in ein Fenster auch in eine Datei plotten lassen. Der gewünschte Input für Gnuplot sieht dann ungefähr so aus:

```
set term table
set samples 201
...
set out "soliton20.dat"
plot [-2*pi:2*pi] besj0(x)**4*A20
set out "soliton21.dat"
plot [-2*pi:2*pi] besj0(x)**4*A21
...
```

3. Automatisches Fitten

In den Dateien »abkling1.dat« bis »abkling368.dat« befinden sich aufgezeichnete Daten von einem mehrfach exponentiell abklingenden Prozess. Fitten Sie jedes Datenfile mit folgender Funktion:

$$f(x) = A \exp(-x) + B \exp(-2x) \quad (5)$$

Geben Sie die Fitparameter A und B zweispaltig in eine Datei aus. Plotten Sie anschließend diese Datei. Was erkennen Sie?

10 Punkte

Hinweis

In Gnuplot können Sie die Ausgabe des Befehls `print` in eine Datei umlenken, um die Fitparameter zu sichern. Die Musterlösung generiert folgende Befehle:

```
set print "xy.dat"
fit a*exp(-x)+b*exp(-2*x) "abkling1.dat" via a,b
print a,b
...
set print
plot "xy.dat"
pause 10
```

Abgabe bis 21. Mai 2006 per E-Mail an Christian.Gollwitzer@uni-bayreuth.de