

# Klausur

## Computergrundlagen WS 2012/2013

JP Dr. Maria Fyta      JP Dr. Axel Arnold      Elena Minina  
Florian Weik      Kai Kratzer      Rudolf Weeber  
Stefan Kesselheim      Tobias Richter

5. März 2013

<b>Name</b>	
<b>Vorname</b>	
<b>Matrikelnummer</b>	

### Hinweise

- Der verfügbare freie Platz gibt einen Hinweis darauf, welchen Umfang die Lösung haben sollte.
- Falls der Platz nicht ausreichen sollte, verwende zusätzliche Blätter. Beschrifte diese unbedingt mit Deinem Namen und Matrikelnummer!
- Einige Fragen ähneln den Fragen aus der Übungsklausur oder vorigen Klausuren, sind aber i.d.R. *nicht* identisch! Lies die Fragen deshalb bitte *genau* durch!
- Die Maximalpunktzahl ist 60.

**Viel Erfolg!**

## 1 Unixgrundlagen (10 Punkte)

**Aufgabe 1:** (1 Punkt)

Was ist der Unterschied zwischen einem Terminal und einer Shell?

**Antwort:**

| Terminal: Tastatur-Eingabe und Zeichen-Ausgabe  
| Shell: startet und verwaltet Programme

**Aufgabe 2:** (1 Punkt)

Gib die Befehlskette an, die ein Verzeichnis `newdir` im aktuellen Verzeichnis erstellt und alle Dateien `oldnotes*` aus dem aktuellen in dieses Verzeichnis kopiert.

**Antwort:**

| `mkdir newdir`  
| `cp oldnotes* newdir/`

**Aufgabe 3:** (1 Punkt)

Gib die Befehle an, um in das Verzeichnis `newdir` zu wechseln und um dir die Dateien in diesem Verzeichnis anzeigen zu lassen.

**Antwort:**

| `cd newdir/`  
| `ls`

**Aufgabe 4:** (1 Punkt)

Das aktuelle Verzeichnis hat folgenden Inhalt:

```
wissen.txt index.txt rezept1.txt rezept3.txt rezept5.txt  
grundrezepte.txt rezept0.txt rezept2.txt rezept4.txt
```

Wie findet man in allen `txt`-Dateien, die mit `rezept` beginnen, diejenigen, die das (Teil-) Wort `Sahne` enthalten?

**Antwort:**

| `grep Sahne rezept*.txt`

**Aufgabe 5:** (1 Punkt)

Was macht der folgende Befehl: „`ln -s $HOME/backup meinbackup`“ ?

**Antwort:**

| Der Befehl erstellt einen symlink im aktuellen Verzeichnis mit dem Namen `meinbackup`, der auf `$HOME/backup` zeigt.

**Aufgabe 6:** (1 Punkt)

Wie findet man alle Dateien im aktuellen Verzeichnis und dessen Unterverzeichnissen, die größer sind als 10MB?

**Antwort:**

```
| find . -size +10M
```

**Aufgabe 7:** (1 Punkt)

Was tut der Befehl „`gedit loesungen.txt &`“ ?

**Antwort:**

```
| Der Befehl ruft das Programm gedit im Hintergrund auf, übergibt  
| „loesungen.txt“ als Argument und kehrt danach zur Shell zurück.
```

**Aufgabe 8:** (1 Punkt)

Was tut der Befehl „`cat notes.txt > /dev/null`“ ?

**Antwort:**

```
| Er leitet den Inhalt der Datei notes.txt nach /dev/null um.
```

**Aufgabe 9:** (1 Punkt)

Was tut folgende Befehlskette und was ist deren Ausgabe?

```
echo "Hallo_Welt" | sed s/Welt/cgl12/g
```

**Antwort:**

```
| Die Ausgabe des echo-Befehls wird über die Pipe an den sed-Befehl gegeben,  
| der das Wort Welt durch cgl12 ersetzt, also Hallo cgl12.
```

**Aufgabe 10:** (1 Punkt)

Hilf Rob aus Abbildung 1, indem du einen gültigen tar-Befehl nennst!

**Antwort:**

```
| tar xf archiv.tar
```

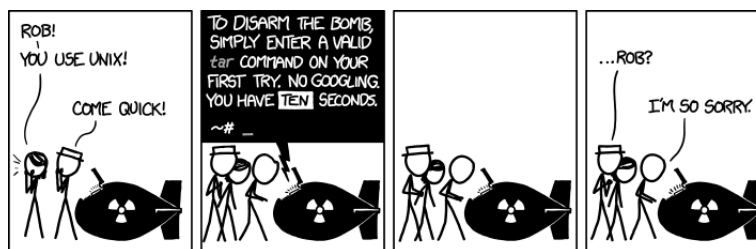


Abbildung 1: <http://xkcd.com/1168/>

## 2 Datei- und Verzeichnisrechte (5 Punkte)

Auf einem Unix-Rechner gibt der Benutzer newton in einer Shell die Befehle `groups newton alice cgl1355 bob` und `ls -la` ein und erhält folgende Ausgabe:

```
$ groups newton alice cgl1355 bob
newton   : icp cgl1112 dozent
alice    : icp cgl1112
cgl1355  : cgl1112
bob      : user
$ ls -la
total 4
drwxrwxr-x  2 newton icp          60 2010-10-27 13:23 .
drwxr-xr-x 22 newton icp        4096 2010-10-27 13:22 ..
-rwxr----- 1 alice  cgl1112      0 2010-10-27 13:23 calculate.sh
-rw-r----- 1 newton icp          0 2010-10-27 13:22 Kraft_auf_Apfel.txt
```

**Aufgabe 11:** (1 Punkt)

Welcher der Benutzer newton, alice, cgl1355 und bob kann die Datei `calculate.sh` löschen?

**Antwort:**

| newton, alice

**Aufgabe 12:** (1 Punkt)

Welcher der Benutzer newton, alice, cgl1355 und bob kann die Datei `calculate.sh` ausführen?

**Antwort:**

| Nur alice

**Aufgabe 13:** (1 Punkt)

Wie sehen die Rechte der Datei `calculate.sh` aus, nachdem Benutzerin alice den Befehl `chmod a+x calculate.sh` ausgeführt hat?

**Antwort:**

| -rwxr-x--x

**Aufgabe 14:** (1 Punkt)

Welcher der Benutzer newton, alice, cgl1355 und bob kann die Datei `Kraft_auf_Apfel.txt` ändern?

**Antwort:**

| Nur newton

**Aufgabe 15:**

(1 Punkt)

Welche Rechte sollte eine Datei `backup.tar.bz2` haben, die ein Backup der persönlichen Daten enthält? Gib einen Befehl an, der diese Rechte setzt.

**Antwort:**

Nur lesbar vom Eigentümer.  
`chmod 600 backup.tar.bz2`

### 3 Python (12 Punkte)

#### Aufgabe 16:

(2 Punkte)

Im folgenden Programm befinden sich 4 Syntaxfehler. Behebe sie. **Hinweis: Der % Operator berechnet den Rest der Division. Z. Bsp.  $10 \% 3 = 1$ .**

```
def function(a,b)
    gefunden=False
    zahl=a
    while not gefunden:
        if (a % zahl = 0) and (b % zahl == 0):
            gefunden=True
        but:
            zahl -= 1
    return zahl

print function(25,20)
```

#### Antwort:

```
1c1
< def ggT(a,b)
---
> def ggT(a,b):
3c3
<     zahl=a
---
>     zahl=a
5c5
<         if (a % zahl = 0) and (b % zahl == 0):
---
>         if (a % zahl == 0) and (b % zahl == 0):
7c7
<         but:
---
>         else:
```

#### Aufgabe 17:

(2 Punkte)

Was tut die Funktion `function(a,b)` aus dem Programm aus der letzten Aufgabe?  
Was passiert, wenn negative Zahlen übergeben werden?

#### Antwort:

Die Funktion berechnet den grössten gemeinsamen Teiler der Zahlen `a` und `b`. Die Funktion liefert nur für `a >= 0` sinnvolle Werte zurück, für negatives `a` bricht die Schleife nie ab.

**Aufgabe 18:**

(2 Punkte)

Ergänze die Funktion `function(a,b)` von oben so, dass für negative Zahlen der Wert 0 zurückgegeben wird.

**Antwort:**

```
def ggT(a,b):  
    if a < 0 or b < 0:  
        return 0  
    [...]
```

**Aufgabe 19:**

(3 Punkte)

Beschreibe in eigenen Worten, was das folgende Programm tut. Was ist die Ausgabe?

```
def f(x):  
    return x*x - 2  
  
def search(a, b):  
    c = 0.5 * (b+a)  
    if f(c) == 0: return c  
    if (b-a) < 1e-6: return c  
  
    if (f(a) > 0 and f(c) < 0) or (f(a) < 0 and f(c) > 0):  
        return search(a,c)  
    else:  
        return search(c,b)  
  
print search(0.,2.)
```

**Antwort:**

Das Programm bestimmt die erste Nullstelle der Funktion  $f(x)$  durch Bisektion. Die Ausgabe ist  $\approx \sqrt{2}$ .

**Aufgabe 20:**

(3 Punkte)

Schreibe ein iteratives Programm, das das selbe tut wie das rekursive Programm aus der vorherigen Aufgabe.

**Antwort:**

```
def f(x):
    return x*x - 2

def search(a, b):
    while (b-a) > 1e-6:
        c = 0.5*(a+b)

        if f(c) == 0:
            break

        if (f(a) > 0 and f(c) < 0) or \
            (f(a) < 0 and f(c) > 0):
            b=c
        else:
            a=c
    return a

print search(0.,2.)
```



## 4 Boole'sche Algebra (2 Punkte)

### Aufgabe 21:

(1 Punkt)

Gegeben sei der Boole'sche Ausdruck  $F = a \wedge (b \wedge a)$ .

Stelle für  $F$  eine Wertetafel mit jeweils allen Belegungen der Variablen  $a$  und  $b$  auf.

### Antwort:

a	b	F
0	0	0
0	1	0
1	0	0
1	1	1

### Aufgabe 22:

(1 Punkt)

Lies aus der Wertetabelle in der vorigen Aufgabe einen vereinfachten Ausdruck für  $F$  ab.

Wie lautet er?

### Antwort:

$a \wedge b$  ergibt dieselbe Tabelle.

## 5 Zahlensysteme (5 Punkte)

### Aufgabe 23: (3 Punkte)

Rechne die Dezimalzahl 49 von Hand (u. U. unter Zuhilfenahme der nebenstehenden Tabelle) in das Hexadezimal-, das Binär- und das Septalsystem (zur Basis 7) um. Gib den Lösungsweg an!

#### Antwort:

$$49 - 16 * 3 = 1 \implies 49_d = 31_h$$

$$27_h = 11\ 0001 \text{ laut Tabelle}$$

$$49 = 1 * 7 * 7 \implies 49_d = 100_7$$

2	7	8	10	16
0	0	0	0	0
1	1	1	1	1
10	2	2	2	2
11	3	3	3	3
100	4	4	4	4
101	5	5	5	5
110	6	6	6	6
111	10	7	7	7
1000	11	10	8	8
1001	12	11	9	9
1010	13	12	10	A
1011	14	13	11	B
1100	15	14	12	C
1101	16	15	13	D
1110	20	16	14	E
1111	21	17	15	F
10000	22	20	16	10

### Aufgabe 24:

(1 Punkt)

Rechne die Hexadezimalzahl BAFF123 in das Binärsystem um. Gib den Lösungsweg an!

#### Antwort:

1011 1010 1111 1111 0001 0010 0011 (Tabelle)

### Aufgabe 25:

(1 Punkt)

Rechne die Binärzahl

1 0 1 0 1 1 0 0 1 1 1 0 0 0 0 0 0 1 1 1 0 0 1 1 0 1 0 1

in das Hexadezimalsystem und Oktalsystem um. Gib den Lösungsweg an!

#### Antwort:

Aus der Tabelle:

1010 1100 1110 0000 0111 0011 0101 = ACE0735<sub>h</sub>

1 010 110 011 100 000 011 100 110 101 = 1263403465<sub>o</sub>

## 6 Fließkommazahlen und numerische Fehler (5 Punkte)

**Aufgabe 26:** (1 Punkt)

Was sind Mantisse, Exponent und Vorzeichen der Fließkommazahl 42.5 (dezimal)?

**Antwort:**

Mantisse: 4.25, Exponent: 1, Vorzeichen: +.

Die Mantisse ist so definiert, dass die führende Stelle  $\neq 0$  ist, nicht die nächstniedrige o. ä. Binär ist dann die führende Stelle immer 1, dezimal 1 – 9.

**Aufgabe 27:** (2 Punkte)

Betrachte das folgende C-Programm:

```
#include <stdio.h>
#include <math.h>
int main() {
    int e = 0;
    double z;

    do {
        z = pow(10.0, e);
        e -= 1;
    } while (z != 0.0);

    printf("ergebnis_10^%d\n", e);
    return 0;
}
```

Wenn der Computer mit reellen Zahlen rechnen könnte, würde dieses Programm anhalten? Was macht dieses Programm aber tatsächlich? Begründe Deine Antwort!

**Antwort:**

Nein, wenn er beliebig kleine, positive Zahlen darstellen könnte, würde das Programm nie enden.

Das Programm probiert aus, was die kleinste, positive reelle Zahl ist, die der Computer mit seinen Fließkommazahlen darstellen kann, und gibt diese aus.

**Aufgabe 28:**

(2 Punkte)

Das folgende Python-Programm soll die Zahlen 0, 0,1, ..., 0,9 ausgeben:

```
x = 0.0
while x < 1.0:
    print x
    x += 0.1
```

Führt man das Programm aus, werden aber trotzdem die Zahlen 0,0, 0,1, ... *bis* 1,0 *einschließlich* ausgegeben, obwohl die Abbruchbedingung der Schleife  $x < 1,0$  ist. Warum? Wie kannst Du das Programm korrigieren?

**Antwort:**

Durch Rundungsfehler wird nicht 1,0 berechnet, sondern 0,999999, was immer noch kleiner als 1 ist. Daher läuft die Schleife noch eins weiter. Die NumPy-Funktion `numpy.arange(0, 1, 0.1)` würde die korrekte Liste erzeugen.

## 7 L<sup>A</sup>T<sub>E</sub>X (8 Punkte)

### Aufgabe 29:

(2 Punkte)

Beschreibe, was die folgenden L<sup>A</sup>T<sub>E</sub>X-Befehle tun.

#### Antwort:

- `\chapter` und `\section`
  - | Starten ein neues Kapitel bzw. einen neuen Abschnitt.
- `\emph`
  - | Markiert eine Hervorhebung
- `\Huge` und `\tiny`
  - | Wählen einen sehr großen bzw. sehr kleinen Zeichensatz

Was machen die folgenden L<sup>A</sup>T<sub>E</sub>X-Umgebungen?

#### Antwort:

- `quote`
  - | Markiert ein Zitat
- `itemize` und `enumerate`
  - | Aufzählungsumgebungen, einmal mit Bullets und einmal durchnummeriert

**Aufgabe 30:**

(2 Punkte)

Was tun die beiden folgenden L<sup>A</sup>T<sub>E</sub>X-Codes?

```
\begin{equation}
%\frac{1+x}{1+y}=A+B*x
\int_{0}^{2\pi} 2\cos(x)\sin(x)dx
\end{equation}
```

```
\begin{figure}
\includegraphics[width=3cm]{grafik.jpg}
\end{figure}
```

**Antwort:**

Der erste Befehl setzt die Gleichung  $\int_0^{2\pi} 2\cos(x)\sin(x)dx$  in einer abgesetzten eigenen Zeile mit Gleichungsnummer. Dabei sind `cos` und `sin` kursiv gesetzt, da der Backslash vergessen wurde, der diese als mathematische Funktionen erklärt (`\cos` bzw. `\cos`).

Der zweite Befehl bindet die Graphik `grafik.jpg` so ein dass sie 3cm breit ist und entsprechend hoch.

**Aufgabe 31:**

(4 Punkte)

Die folgende Tabelle wurde mit L<sup>A</sup>T<sub>E</sub>X gesetzt:

7DD	hexadezimal
3735	oktal
11111011101	binär
2013	dezimal

Schreibe einen L<sup>A</sup>T<sub>E</sub>X-Code, der diese Tabelle erzeugen würde.

**Antwort:**

```
\begin{tabular}{r|c} \hline
7DD & hexadezimal \\ \hline
3735 & oktal \\ \hline
11111011101 & bin\"ar \\ \hline \hline
2013 & dezimal
\end{tabular}
```

## 8 Bildbearbeitung (2 Punkte)

### Aufgabe 32:

(1 Punkt)

Nenne je ein Dateiformat, das Bilder als Pixel- und Vektorgrafik speichert.

### Antwort:

PNG oder JPEG sind Pixelgrafikformate, EPS oder SVG Vektorgrafikformate.

### Aufgabe 33:

(1 Punkt)

Was ist der Unterschied zwischen einer Vektorgrafik und einer Pixelgrafik? Wann verwendet man das eine, wann das andere?

### Antwort:

Die Vektorgrafik speichert Objekte, die durch bestimmte Punkte und Eigenschaften beschrieben werden, etwa die Ecken und Farben eines Rechtecks. Das ist immer von Vorteil, sofern die Daten entsprechend vorliegen, etwa beim Zeichnen eines Graphen. Eine Pixelgrafik enthält hingegen eine feste Anzahl von Rechtecken (Pixeln), für die jeweils nur die Farbe gespeichert werden muss. Das ist für Photos besser geeignet, weil diese durch Pixelsensoren entstehen.

## 9 Gnuplot (4 Punkte)

### Aufgabe 34:

(2 Punkte)

Beschreibe, was die folgenden Gnuplot-Kommandos tun:

```
set xrange [-5:5]
set xlabel 'x'
set ylabel 'f(x)'
plot x**2+x*sin(x) w lp
```

### Antwort:

Im Bereich von  $x=-5$  bis  $5$  wird die in der Datei  $x**2+x*\sin(x)$  mit Punkten gezeichnet, die durch Linien verbunden sind. Die x-Achse wird mit  $x$  beschriftet,  $y$  mit  $f(x)$ . Die Funktion selber wird in der Legende mit dem Dateinamen  $x**2+x*\sin(x)$  bezeichnet.

### Aufgabe 35:

(2 Punkte)

Die Datei `daten.dat` enthalte in 2 Spalten die Daten einer Simulation. Beschreibe die Ausgabe der folgenden Gnuplot-Kommandos. Insbesondere beschreibe, wozu der Befehl `fit` dient.

```
f(x) = a*exp(-x*b)
fit f(x) "daten.dat" via a,b
plot "datei.dat" with points title "data", \
      f(x) title "a*exp(-x*b) "
```

### Antwort:

Die Daten werden an eine einfache exponentielle Funktion der Form  $a \exp(-bx)$  gefittet. Das Plot-Kommando zeichnet die Daten gegen die gefittete Funktion.



## 10 C (7 Punkte)

### Aufgabe 36:

(4 Punkte)

Das Integral

$$\int_a^b f(x) dx$$

kann angenähert werden durch die Summe

$$I_N = h \sum_{i=0}^{N-1} f(a + ih)$$

mit  $h = \frac{b-a}{N}$ . Schreibe ein C-Programm, das diese Summe für die Funktion

$$f(x) = x^2 - 1$$

berechnet, mit  $a = 0$  und  $b = 2$ . Ergänze dazu im folgenden Quelltext die Funktionen `integral` und `f` so das erstere die gewünschte Näherung zurück gibt.

**Antwort:**

```
double f(double x) {
    return x*x - 1;
}

double integral(void) {
    double a=0; double b=2;
    int N=100; double I=0;
    double h = (b-a)/N; // Schrittweite
    // Schleife, um die Summe zu berechnen
    for (int i = 0; i < N; ++i) {
        I += f(a + i*h);
    }
    // mit der Schrittweite multiplizieren
    I *= h;

    return I; }
```

**Aufgabe 37:**

(1 Punkt)

Schreibe eine Funktion `void verdopple(double *p)`, die einen Zeiger auf einen `double`-Wert erhält, und den Wert an der entsprechenden Stelle im Speicher verdoppelt.

**Antwort:**

```
void verdopple(double *p) {
    (*p) *= 2;
}
```

**Aufgabe 38:**

(2 Punkte)

Benenne einen Vorteil, den C gegenüber Python besitzt, und einen Vorteil den Python gegenüber C besitzt.

**Antwort:**

C ist schnell.  
Python kommt mit vielen Bibliotheken.