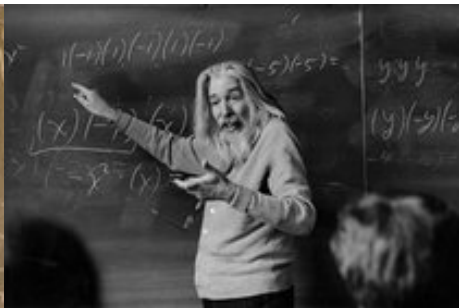
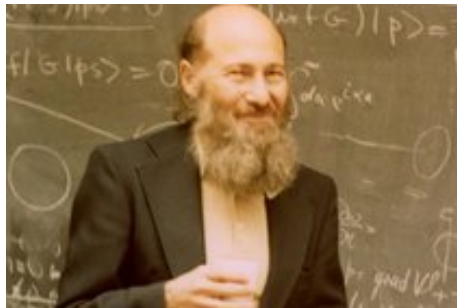


Computer und Mathematik

Stand: 12. Oktober 2017

Wozu braucht ein Mathematiker einen Computer???



Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen

Wozu braucht ein Mathematiker einen Computer???



- Zum Rechnen
Rechner und Rechnerinnen der NASA

Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen
- Zum Schreiben von Büchern und anderen Veröffentlichungen

Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen
- Zum Schreiben von Büchern und anderen Veröffentlichungen
- Zum Visualisieren von mathematischen Objekten

Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen
- Zum Schreiben von Büchern und anderen Veröffentlichungen
- Zum Visualisieren von mathematischen Objekten
- Zum Beweisen von Sätzen

Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen
- Zum Schreiben von Büchern und anderen Veröffentlichungen
- Zum Visualisieren von mathematischen Objekten
- Zum Beweisen von Sätzen
- Als mathematisches Objekt

Wozu braucht ein Mathematiker einen Computer???

- Zum Rechnen
- Zum Schreiben von Büchern und anderen Veröffentlichungen
- Zum Visualisieren von mathematischen Objekten
- Zum Beweisen von Sätzen
- Als mathematisches Objekt
- ...

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen

Wozu braucht ein Mathematiker einen Computer???

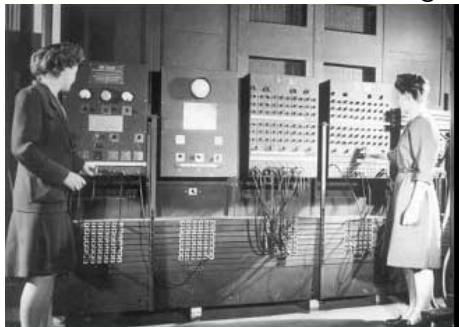
Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen



B. Jennings und F. Bilas an der ENIAC

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen
- Zahlentheorie

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Nahrungweise Losung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflachen
- 1946: ENIAC: Ballistik von Schiffsgeschutzen



- Zahlentheorie
D.H. Lehmer mit der ENIAC

Wozu braucht ein Mathematiker einen Computer???

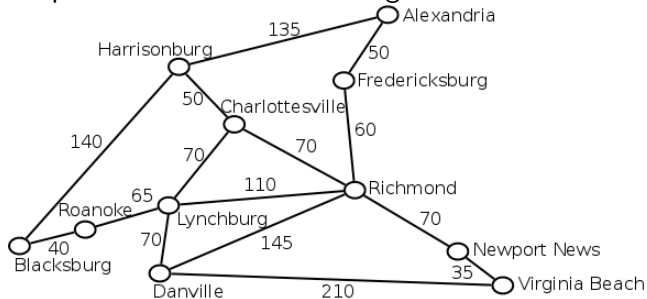
Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen
- Zahlentheorie
- Graphentheorie: bestimmte Wege in Netzen finden

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Nährungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen
- Zahlentheorie
- Graphentheorie: bestimmte Wege in Netzen finden

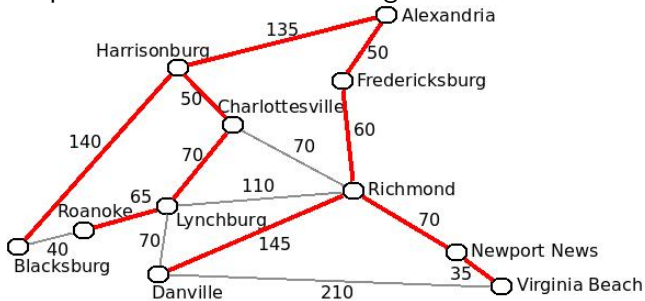


Ein Graph

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Nährungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen
- Zahlentheorie
- Graphentheorie: bestimmte Wege in Netzen finden



Ein Spanning Tree in dem Graph

Wozu braucht ein Mathematiker einen Computer???

Zum Rechnen

- Numerik: Näherungsweise Lösung von Gleichungssystemen
- 1941: Konrad Zuses Z3: Eigenschaften von Tragflächen
- 1946: ENIAC: Ballistik von Schiffsgeschützen
- Zahlentheorie
- Graphentheorie: bestimmte Wege in Netzen finden
- N.B. nicht die schnelleren Rechner bringen, sondern die schnelleren Algorithmen

Ein Beispiel (hier mit dem Programm Sage)

```
A = random_matrix(QQ,12); show(A)
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -2 & -4 & -7 & -13 & 38 & 55 & 10 \\ -1 & 1 & 1 & 2 & -2 & 5 & 3 & 3 & 7 & -26 & -36 & -12 \\ -2 & 0 & 1 & 3 & -2 & 8 & 5 & 4 & 12 & -50 & -40 & -55 \\ 1 & -1 & -1 & -1 & 1 & -2 & -1 & -4 & -9 & 31 & 19 & 42 \\ -2 & 1 & 1 & 3 & -2 & 7 & 2 & 0 & 4 & -28 & -4 & -55 \\ 0 & 0 & -1 & -2 & 1 & -4 & -2 & -1 & 0 & 7 & 17 & -11 \\ 0 & 2 & 1 & 3 & -4 & 5 & 3 & 2 & 2 & -1 & -53 & 57 \\ 0 & -1 & 0 & -1 & 1 & 0 & 0 & 0 & 1 & -11 & 14 & -37 \\ 0 & 0 & 1 & 4 & -2 & 8 & 5 & 1 & 1 & -11 & -42 & 34 \\ 0 & 0 & 0 & -2 & 2 & -1 & 2 & 5 & 14 & -41 & -24 & -55 \\ -1 & 0 & 0 & 0 & 0 & -2 & -4 & -5 & -6 & 14 & 57 & -44 \\ 0 & 0 & 0 & 1 & -1 & 4 & 5 & 4 & 9 & -29 & -47 & -1 \end{pmatrix}$$

Ein Beispiel (hier mit dem Programm Sage)

```
A = random_matrix(QQ,12); A.echelonize(); show(A)
```

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Das wollen Sie nicht von Hand ausrechnen!

Wozu braucht ein Mathematiker einen Computer???

Zum Schreiben

- Mathematische Texte neigen dazu, flächige Elemente zu haben

Wozu braucht ein Mathematiker einen Computer???

Zum Schreiben

- Mathematische Texte neigen dazu, flächige Elemente zu haben

$$\frac{1}{\pi} = \frac{\sqrt{8}}{9801} \sum_{n=0}^{\infty} \frac{(4n)! (1103 + 26390n)}{(n!)^4 396^{4n}}$$

Wozu braucht ein Mathematiker einen Computer???

Zum Schreiben

- Mathematische Texte neigen dazu, flächige Elemente zu haben

$$\frac{1}{\pi} = \frac{\sqrt{8}}{9801} \sum_{n=0}^{\infty} \frac{(4n)! (1103 + 26390n)}{(n!)^4 396^{4n}}$$

- Deswegen gibts ein extra Textsatzprogramm, das sich zu lernen lohnt:

L^AT_EX

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben
- Damit kann man Umformen, Integrieren, Ableiten, etc automatisieren und die Software kann das auch mit recht grossen Ausdrücken

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben
- Damit kann man Umformen, Integrieren, Ableiten, etc automatisieren und die Software kann das auch mit recht grossen Ausdrücken
- Man kann Gegenbeispiele vom Rechner suchen lassen

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben
- Damit kann man Umformen, Integrieren, Ableiten, etc automatisieren und die Software kann das auch mit recht grossen Ausdrücken
- Man kann Gegenbeispiele vom Rechner suchen lassen
- Es gibt *Proof Assistents* als Software

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben
- Damit kann man Umformen, Integrieren, Ableiten, etc automatisieren und die Software kann das auch mit recht grossen Ausdrücken
- Man kann Gegenbeispiele vom Rechner suchen lassen
- Es gibt *Proof Assistents* als Software
- Gegeben Axiome und Schlussregeln kann der Proof Assistant helfen, die nötigen Voraussetzungen für Sätze zu finden.

Wozu braucht ein Mathematiker einen Computer???

Zum Beweisen

- Schon in den 60ern wurde Software für symbolische Algebra geschrieben
- Damit kann man Umformen, Integrieren, Ableiten, etc automatisieren und die Software kann das auch mit recht grossen Ausdrücken
- Man kann Gegenbeispiele vom Rechner suchen lassen
- Es gibt *Proof Assistents* als Software
- Gegeben Axiome und Schlussregeln kann der Proof Assistant helfen, die nötigen Voraussetzungen für Sätze zu finden.
- Eine moderne Richtung in der Mathematik versucht alle Ergebnisse automatisch prüfbar zu formulieren (*Univalent Foundations*)

Umformelungen an einem Beispiel (mit Sage ermittelt)

Die Helligkeit am Punkt x zwischen zwei Lichtquellen, die mit Leuchtkraft P_1, P_2 , im Abstand s , von Höhen h_1, h_2 leuchten:

$$C(x) = \frac{P_1 h_1}{(h_1^2 + x^2)^{\frac{3}{2}}} + \frac{P_2 h_2}{(h_2^2 + (s - x)^2)^{\frac{3}{2}}}$$

Umformelungen an einem Beispiel (mit Sage ermittelt)

Ableitung davon

$$C'(x) = \frac{3 P_2 h_2 (s - x)}{\left(h_2^2 + (s - x)^2\right)^{\frac{5}{2}}} - \frac{3 P_1 h_1 x}{\left(h_1^2 + x^2\right)^{\frac{5}{2}}}$$

Umformelungen an einem Beispiel (mit Sage ermittelt)

Ableitung davon

$$C'(x) = \frac{3 P_2 h_2 (s - x)}{(h_2^2 + (s - x)^2)^{\frac{5}{2}}} - \frac{3 P_1 h_1 x}{(h_1^2 + x^2)^{\frac{5}{2}}}$$

Auf Hauptnenner gebracht, ist der Zähler der rechten Seite

$$\begin{aligned} & -9 P_2^2 h_1^{10} h_2^2 s^2 + 18 P_2^2 h_1^{10} h_2^2 s x - 18 (5 P_1^2 h_1^2 - P_2^2 h_2^2) s x^{11} + 9 (P_1^2 h_1^2 - P_2^2 h_2^2) x^{12} + \\ & 9 (5 (P_1^2 - P_2^2) h_1^2 h_2^2 + (45 P_1^2 h_1^2 - P_2^2 h_2^2) s^2) x^{10} - 90 (12 P_1^2 h_1^2 s^3 + (4 P_1^2 - P_2^2) h_1^2 h_2^2 s) x^9 - \\ & 45 (2 P_2^2 h_1^4 h_2^2 - 2 P_1^2 h_1^2 h_2^4 - 42 P_1^2 h_1^2 s^4 - (28 P_1^2 - P_2^2) h_1^2 h_2^2 s^2) x^8 - \\ & 36 (70 P_1^2 h_1^2 h_2^2 s^3 + 63 P_1^2 h_1^2 s^5 - 5 (P_2^2 h_1^4 h_2^2 - 3 P_1^2 h_1^2 h_2^4) s) x^7 - \\ & 90 (P_2^2 h_1^6 h_2^2 - P_1^2 h_1^2 h_2^6 - 35 P_1^2 h_1^2 h_2^2 s^4 - 21 P_1^2 h_1^2 s^6 + (P_2^2 h_1^4 h_2^2 - 15 P_1^2 h_1^2 h_2^4) s^2) x^6 - \\ & 180 (10 P_1^2 h_1^2 h_2^4 s^3 + 14 P_1^2 h_1^2 h_2^2 s^5 + 6 P_1^2 h_1^2 s^7 - (P_2^2 h_1^6 h_2^2 - 2 P_1^2 h_1^2 h_2^6) s) x^5 - \\ & 45 (P_2^2 h_1^8 h_2^2 - P_1^2 h_1^2 h_2^8 - 30 P_1^2 h_1^2 h_2^4 s^4 - 28 P_1^2 h_1^2 h_2^2 s^6 - 9 P_1^2 h_1^2 s^8 + 2 (P_2^2 h_1^6 h_2^2 - 6 P_1^2 h_1^2 h_2^6) s^2) x^4 - \\ & 90 (4 P_1^2 h_1^2 h_2^6 s^3 + 6 P_1^2 h_1^2 h_2^4 s^5 + 4 P_1^2 h_1^2 h_2^2 s^7 + P_1^2 h_1^2 s^9 - (P_2^2 h_1^8 h_2^2 - P_1^2 h_1^2 h_2^8) s) x^3 - \\ & 9 (P_2^2 h_1^{10} h_2^2 - P_1^2 h_1^2 h_2^{10} - 10 P_1^2 h_1^2 h_2^6 s^4 - 10 P_1^2 h_1^2 h_2^4 s^6 - 5 P_1^2 h_1^2 h_2^2 s^8 - P_1^2 h_1^2 s^{10} + 5 (P_2^2 h_1^8 h_2^2 - P_1^2 h_1^2 h_2^8) s^2) x^2 \end{aligned}$$

Das möchten Sie nicht von Hand aus-x-en

Wozu braucht ein Mathematiker einen Computer???

Als mathematisches Objekt

- Rechner wurden schon mathematisch formalisiert, bevor sie automatisiert wurden (Turing und Church 1936)

Wozu braucht ein Mathematiker einen Computer???

Als mathematisches Objekt

- Rechner wurden schon mathematisch formalisiert, bevor sie automatisiert wurden (Turing und Church 1936)
- **Theorie der Berechenbarkeit** Welche mathematischen Objekte „passen“ überhaupt in einen Rechner?

Wozu braucht ein Mathematiker einen Computer???

Als mathematisches Objekt

- Rechner wurden schon mathematisch formalisiert, bevor sie automatisiert wurden (Turing und Church 1936)
- **Theorie der Berechenbarkeit** Welche mathematischen Objekte „passen“ überhaupt in einen Rechner?
- **Komplexitätstheorie** Wie lange braucht ein Lösungsverfahren schlimmstenfalls oder im Durchschnitt, um ein Problem gegebener Grösse zu lösen?

Wozu braucht ein Mathematiker einen Computer???

Als mathematisches Objekt

- Rechner wurden schon mathematisch formalisiert, bevor sie automatisiert wurden (Turing und Church 1936)
- **Theorie der Berechenbarkeit** Welche mathematischen Objekte „passen“ überhaupt in einen Rechner?
- **Komplexitätstheorie** Wie lange braucht ein Lösungsverfahren schlimmstenfalls oder im Durchschnitt, um ein Problem gegebener Grösse zu lösen?
- **Korrektheit** Wie kann man mathematisch beweisen, dass ein gegebenes Programm das tut, was es (laut formaler Spezifikation) tun soll?

Wozu braucht ein Mathematiker einen Computer???

Als mathematisches Objekt

- Rechner wurden schon mathematisch formalisiert, bevor sie automatisiert wurden (Turing und Church 1936)
- **Theorie der Berechenbarkeit** Welche mathematischen Objekte „passen“ überhaupt in einen Rechner?
- **Komplexitätstheorie** Wie lange braucht ein Lösungsverfahren schlimmstenfalls oder im Durchschnitt, um ein Problem gegebener Grösse zu lösen?
- **Korrektheit** Wie kann man mathematisch beweisen, dass ein gegebenes Programm das tut, was es (laut formaler Spezifikation) tun soll?
- Diese mathematische Fragestellung hat zur Entwicklung von Programmiersprachen geführt, die näher an der Mathematik als an der Hardware sind, zum Beispiel ML und Haskell

Was bietet die Uni?

Das Rechenzentrum

Was bietet die Uni?

Das Rechenzentrum

- stellt WiFi (<https://www.anleitungen.rrze.fau.de/internet-zugang/wlan/>)
- bietet mit FAUBox einen lokalen DropBox Clone an (<https://www.rrze.fau.de/serverdienste/server/faubox/>)
- verwaltet Accounts zentral (idm.fau.de)

Was bietet die Uni?

Das Rechenzentrum

- stellt WiFi (<https://www.anleitungen.rrze.fau.de/internet-zugang/wlan/>)
- bietet mit FAUBox einen lokalen DropBox Clone an (<https://www.rrze.fau.de/serverdienste/server/faubox/>)
- verwaltet Accounts zentral (idm.fau.de)

Das Department Mathematik

Was bietet die Uni?

Das Rechenzentrum

- stellt WiFi (<https://www.anleitungen.rrze.fau.de/internet-zugang/wlan/>)
- bietet mit FAUBox einen lokalen DropBox Clone an (<https://www.rrze.fau.de/serverdienste/server/faubox/>)
- verwaltet Accounts zentral (idm.fau.de)

Das Department Mathematik

- betreibt Rechnerräume zu Ihrer Verfügung: PC Pool 1 (Raum 00.230) und 2 (Raum 00.326), Praktikum 1 (Raum 00.325)
- bietet Mathematikstudenten ein Drucker-Freikontingent pro Semester (aus Studienzuschussmitteln) (<https://math.fau.de/drucker>)
- stellt Hilfskräfte ein, an die Sie sich wenden können:
problems@math.fau.de, Telefon 67335 von den Telefonen in den Rechnerräumen

Arbeiten mit den Rechnern des D. Mathematik

- Login mit dem RRZE Account (der Form ab12cdef)
- Die Maschinen laufen unter Linux
- Sie können beim Einloggen zwischen vielen graphischen Oberflächen wählen, probieren Sie mehrere!
- Die Maschinen sind identisch konfiguriert
- Es sind Unmengen an Programmen installiert
- Sie können ein Freidruckkontingent beantragen und die Drucker in den Räumen damit benutzen
- Sie haben 1.5 Gigabyte Plattenplatz für sich, das HOME
- Wenn Sie diese Quota überschreiten, **können Sie sich (erstmal) nicht mehr einloggen**

Arbeiten mit den Rechnern des Dep. Mathematik

- Regel eins: **Haben Sie keine Angst, irgendwas aus Versehen kaputt machen zu können**
- Es gibt Backups aller HOMEs
- Das Betriebssystem wird Sie dran hindern, andere zu stören

- Unsere Arbeitsplätze laufen unter Linux.
- Linux versucht, Unix zu clonen.
- Unix war ein Betriebssystem, dessen Design-Ziel es war, alles mit einfachen Werkzeugen zu erledigen: Input/Output war zumeist Text. Gebaut für Programmierer.
- Login hier mit dem RRZE-Usernamen und -Passwort (siehe Immatrikulationsunterlagen)
- Unter dem Passwort-Fenster kann man das Tastaturlayout einstellen, falls das nicht zur Tastatur passt.
- Wenn etwas nicht klappt, bitte mitteilen
- Jeder User hat einen HOME-Ordner, wo er/sie Daten ablegen kann.
- In das HOME dürfen maximal 1.5 Gigabyte!

Probieren Sie,

- sich einzuloggen

- Es zeigt sich nach dem Login erst mal eine äußerst bunte Oberfläche.
- Diese Oberflächen sehen alle naselang anders aus
- Unter den diversen Oberflächen gibts aber die **Shell** („Terminal“, „Konsole“, ...)
- Die Shell ist ein Text-Interface zum Betriebssystem und zahlreichen Programmen. Ein Teil dieser Programme ist standardisiert und sollte über lange Zeit und jede Variante von UNIX/Linux gleich sein. Das heißt: einmal gelernt – und das hält lang.
- Die Shell funktioniert auch aus der Ferne, wenn man sich von einem System aus auf einem anderen einloggt.
- Man kann aus Shell-Kommandos Skripte (kleine Programme) schreiben, die Aufgaben automatisch erledigen.

- Um eine Shell in einer Gnome-Umgebung zu kriegen, Strg+Alt+T drücken.
- Dort erwartet Sie ein „Prompt“.
- Danach kann man zeilenweise Kommandos und deren Parameter eingeben.
- Struktur ist immer `kommando argument1 argument2 ...`, durch Leerzeichen getrennt, Eingabe wird mit der Eingabetaste beendet.
- Zu den meisten Kommandos gibt es Dokumentation (sog *Man-Pages*) im System selbst, die man mit dem `man`-Befehl kriegt, z. B. `man sh` (Man-Page der Shell selbst), `man man` (Man-Page für `man`)
- Eine kleine Einführung gibt `man intro`.
- In den Man-Pages kann man mit Leertaste vorwärts, mit `b` rückwärts blättern, mit `q` beenden.

Probieren Sie,

- ein Terminal zu öffnen
- `man intro` zu rufen. Lassen Sie dieses Fenster mit der Man-Page offen.
- in `manpages` kann man mit `SPACE` vorwärts-, mit `b` rückwärts blättern
- mit `q` beendet man das Textblätternprogramm (*pager*)

Bewegung und Übersicht im Dateisystem:

`ls` **LiStet** die Dateien und Verzeichnisse im aktuellen Ordner

`ls -l` Listet die Dateien und Verzeichnisse im aktuellen Ordner zusammen mit Infos über Größe, Zeitpunkt der letzten Änderung, Besitzer, Zugriffsrechte und anderes.

`ls /` Listet die Dateien und Verzeichnisse im obersten Verzeichnis.

`pwd` **PrintWorkingDirectory**: Wo bin ich im Dateibaum?

`cd /usr/share/dict` **ChangeDirectory**: Gehe woandershin.

`cd` (ohne Argument): Gehe ins eigene HOME.

`cd ..` Gehe ein Verzeichnis höher
z. B. `pwd`

Probieren Sie,

- mit `ls` und `ls -l` herauszufinden, welche Dateien und Ordner in Ihrem HOME liegen.
- die `-a` Option an das Kommando `ls` dranzuhängen. Was tut es?
- mit `cd` in das Verzeichnis `/home/stud` zu wechseln.
- dort mal `ls` einzugeben.
- mit `cd` nach `/usr/bin` zu gehen und sehen Sie nach, welche Dateien dort liegen.

Die meisten Programme aus der UNIX-Tradition arbeiten auf Textdateien.
Textdateien kann man mit einem Editor erstellen und verändern.

Es gibt verschiedenste Editoren:

- Urgestein: `vi`
- Für Gurus: `emacs`
- Im Terminal, aber einfach: `nano`
- Graphisch: `gedit` (und viele andere)

Probieren Sie,

- einen Editor Ihrer Wahl zu starten
- damit eine Textdatei von einigen Zeilen Länge zu erzeugen
- diese in Ihrem HOME zu speichern

- Sobald man Dateien hat, kann man mit ihnen was machen, z. B.
 - ▶ `cd /usr/share/dict; wc -l words` **WordCount**. Das „-l“ ist eine *Option*. Damit wird nur die Anzahl der Zeilen (engl. *lines*) angezeigt. Mehrere Kommandos in der selben Zeile kann man mit Strichpunkt ; trennen.
 - ▶ `rm datei` **ReMove**: löscht die Datei. Es gibt **kein** `unremove`, **kein** Trash oder so was.
 - ▶ `mv datei anderedatei` **MoVe**: benennt die Datei um
 - ▶ `cp datei nochnedatei` **CoPy**: kopiert die Datei

Probieren Sie,

- Die Datei, die Sie eben erzeugt haben, mit `cp` zu kopieren. Prüfen Sie mit `ls`, ob das geklappt hat.
- Ihre Datei so umzubenennen, das ihr neuer Name ein Leerzeichen enthält. Was bemerken Sie?
- eine Kopie Ihrer Datei zu löschen.
- eine Datei in `/usr/bin` zu löschen. Warum geht das nicht?

Pipes

- Man kann Kommandos „kombinieren“, indem man den Output des ersten als Input eines zweiten verwendet:
`grep "^a" /usr/share/dict/words | wc -l` liefert die Anzahl aller Worte in `words`, die mit `a` anfangen
- Das `|` heisst *Pipe*.
- Wenn man einem Programm keinen Dateinamen mitgibt (wie in `wc -l`), dann liest es vom „*standard input*“. Das ist entweder die Tastatur (bzw. der „Einfüge-Puffer“) oder der Output eines anderen Programms, das mit einer *Pipe* verbunden ist.
- Nützlich, um Ausgaben lesbar zu halten:
`grep "^a" /usr/share/dict/words | less` zeigt die gefundenen Wörter – der Output wird jedoch seitenweise angezeigt und man kann blättern (mit `SPACE` und `b`, beenden mit `q`, suchen mit `/`)
- Merke: `h a | g | f` ist mathematisch geschrieben $f(g(h(a)))$

Probieren Sie,

- eine Ihrer Dateien durch die Programme `tac` und `rev` zu schicken

- `du -k datei` **DiskUsage** liefert die Größe einer Datei in Kilobyte
- `du -ak` (`-a = all`) liefert die Grösse aller Dateien im aktuellen Verzeichnis
- `du -ka | sort -n` sortiert den Output von `du -ka` numerisch nach Größe
- Man kann den Output eines Kommandos auch in eine Datei umleiten:
`du -ak > /tmp/allemeinefiles`
- Analog auch den Input eines Kommandos aus einer Datei holen:
`sort -n < /tmp/allemeinefiles`
- `sed` ist der **StreamEDitor**, mit dem man u. a. solche Sachen machen kann wie `sed 's/regex/wort/' < file`, was jedes Vorkommen des regulären Ausdrucks `regex` durch `wort` ersetzt, den restlichen Text aber unverändert lässt.
- `comm f1 f2` findet in den (sortierten) Dateien die Zeilen, die
 - 1 – nur in der Datei `f1` (links angeordnet)
 - 2 – nur in der Datei `f2` (mittig)
 - 3 – in beiden Dateien (rechts)

vorkommen

Probieren Sie,

- herauszufinden, wie viele Dateien in Ihrem HOME-Ordner liegen
- den Inhalt der von Ihnen erzeugten Dateien alphabetisch zu sortieren (`man sort` hilft)

- `diff f1 f2` liefert die textuelle „Differenz“ zwischen zwei Texten, d. h. die Zeilen in der Datei `f1`, die man ändern muss, um zum Text in der Datei `f2` zu gelangen.
- Hinweis: wenn man einen Dateinamen tippt und bereits so viele Zeichen eingegeben hat, dass im jeweiligen Verzeichnis nur noch eine Datei mit der passenden Zeichenkette existiert, kann man mit der TAB-Taste den Input automatisch vervollständigen lassen.
- Ein Programm, das nicht zu reagieren scheint, kann man mit der Tastenkombination `Ctrl+C` abbrechen.
- Wenn man einem Programm über den *standard input* etwas zukommen lassen will, kann man die Eingabe mit `Ctrl+D` abschicken. Das Programm verarbeitet die abgeschickte Zeichenkette (und beendet sich danach).
- Die Shell ist selbst so ein Programm, d. h. `Ctrl+D` im Prompt beendet die Shell.

Probieren Sie,

- eine Ihrer Textdateien zu kopieren und die Kopie zu verändern
- die Unterschiede der beiden Dateien mittels `diff` zu erfassen
- sortierte Kopien der beiden Dateien unter den Namen `x.s` und `y.s` zu erzeugen
- mittels `comm` diejenigen Zeilen auszugeben, die nur in `x.s` existieren
Warum arbeitet `comm` wohl nur auf sortierten Dateien?

Mal was Anspruchsvolleres

- Lesen Sie die Man-Page zum Kommando `tr` und implementieren Sie damit anschließend die `rot13`-Verschlüsselung, bei der `a` auf `n`, `b` auf `o`, `c` auf `p`, ... und jeweils umgekehrt abgebildet werden, und entschlüsseln Sie den Ciphertext `guvax`.
- Für die ganz Harten: Lesen Sie die Man-Pages von `comm` und `sort` und finden Sie diejenigen Zeilen in `/usr/share/dict/words`, die als Output einer `rot13`-Verschlüsselung von `words` entstehen können.

Weiterführendes

- Informatiker Linux-Kurs <https://fsi.cs.fau.de/linuxkurs>
- WikiBook zu Shell Programmierung
https://en.wikibooks.org/wiki/Bash_Shell_Scripting
- Die Bibel, geschrieben vom Papst:
"The UNIX Programming Environment" von B. Kernighan

