ulm university universität
uulm

Facetten der Informatik

Skript und Materialien für den
VIMP-Kurs 2008/09

Guido de Melo

2008/09

Fassung 30. Oktober 2009

Satz: PDF-LaTeX 2ε

Inhaltsverzeichnis

1 Vorwort

Schon seit einigen Jahren gibt es an der Universität Ulm einen Kurs zur Förderung begabter Schüler, den VIMPs-Kurs. Der Kurs gliedert sich in mehrere Teile: Mathematik, Informatik und Physik und soll interessante Inhalte aus diesen Gebieten vermitteln. Früher lag der Schwerpunkt der Informatik auf der Erstellung von Webseiten. Dabei konnten nur wenige Themen der Informatik behandelt werden. Mittlerweile habe ich den Kurs umgestellt um einige Aspekte der Informatik die jedem häufig begegnen zu zeigen.

Ich bin Guido de Melo, derzeit noch Doktorand in der Fakultät für Ingenieurwissenschaften und Informatik. Anfang 2010 werde ich die Prüfung zum Doktor ablegen. Ich habe schon verschiedene Förderprogramme für Schüler an der Uni begleitet, da ich es sehr wichtig finde dem Interessierten Themen aus meinem Bereich näher zu bringen.

Diese kleine Schrift ist eine Kombination aus Skript und einer Sammlung von Aufgaben und Material zu verschiedenen Themen aus der Informatik. Sie handelt von so grundlegenden Dingen wie dem Begriff des Algorithmus genauso wie von aktuellen Themen wie Datenschutz und Privatsphäre oder der Kompression von Daten in Bildern oder Videos, also von Dingen die uns täglich begegnen und von denen wir direkt oder indirekt betroffen sind.

Ich freue mich, wenn man mit mir Kontakt aufnimmt. Sei es wegen gefundenen Fehlern in diesem Skript, Anregungen oder auch aus Interesse. Ich kann aber keine Hilfe bei Problemen mit dem Rechner geben, würde ich das tun, so wäre ich auf längere Zeit ausgebucht ;-)

Zu erreichen bin ich unter `guido.de-melo@alumni.uni-ulm.de` oder über meine Webseite `http://guido.demelo.de`.

Viel Spaß beim Lesen und Lernen!
Guido de Melo

2 Was ist Informatik

Zum Start der Veranstaltung beschäftigen wir uns allgemein mit der Informatik. Wir werden untersuchen was Informatik ausmacht und wenden uns erst später Maple zu.

2.1 Felder der Informatik

Die Informatik ist eine Strukturwissenschaft die sich im Kern mit Daten, Strukturen in Daten und der Verarbeitung von Daten befasst. Dabei erforscht sie Verfahren effizient mit großen Datenmengen zu arbeiten und schwierige Berechnungen anzustellen. Sie umfasst eine Reihe von Feldern.

Feld	**Beispiele**
Eingebettete Systeme	Waschmaschine, DVD-Player
Datenbanken	Telefonauskunft, Warenkatalog
Medien	Video, Audio, interaktive Systeme
Verteilte Systeme	World Wide Web
Künstliche Intelligenz	Planungssysteme
Neuroinformatik	Bildverarbeitung

2.2 Was bringt Informatik?

Schreibe auf was Informatik

a) für die Wissenschaft
b) für die Wirtschaft
c) für die Technik
d) für die Bildung
e) für die Kultur
f) für den Einzelnen und die Gesellschaft

bringt. Deine Gruppe soll dabei einige Vorteile sammeln und danach gewichten welche drei ihr als am wichtigsten seht. Schreibt die drei wichtigsten Vorteile und warum ihr gerade diese gewählt habt in je einem Satz auf.

2.3 Geschichte

Was sind die wichtigsten Meilensteine in der Informatik? Nenne eine Hand voll Meilensteine und warum gerade diese so wichtig sind. Stimme dich mit der Gruppe zu den berühmten Personen der Informatik ab.

1673 Vier-Spezies Maschine
1805 Lochkarte
1833 Analytical Engine
1843 Erstes Programm von Ada Lovelace
1937 mechanische Z1 von Zuse
1941 Z3 mit Relais
1944 Mark 1 in den USA (John von Neumann)

1946 ENIAC
1956 formale Grammatiken
1957 Fortran
1959 Lisp
1959 Cobol
1960 Algol, Basic
1966 ELIZA
1970 C
1971 Smalltalk
1972 Prolog
1973 Xerox Alto (GUI, Maus, Wimp)
1976 Apple 1
1976 SQL
1982 C64
1986 i386
1990 C++
1995 Java
2000 C#

2.4 Berühmte Personen der Informatik

Informatik als Fach gibt es noch nicht so lange, doch Gedanken zur Informatik haben sich schon vor längerer Zeit Leute gemacht. Greife ein paar besonders wichtige Personen heraus und erzähle kurz warum gerade diese wichtig sind. Stimme dich mit der Gruppe zur Geschichte ab.

1646-1716 Gottfried Wilhelm Leibnitz
1791-1871 Charles Babbage
1815–1852 Ada Lovelace
1815-1864 George Bool
1912-1954 Alan Turing
1910-1995 Konrad Zuse
1903–1957 John von Neumann
1900-1973 Howard Aiken
1906-1992 Grace Hopper
1923-2008 Joseph Weizenbaum
1928- Noam Chomsky

2.5 Informatik als Naturwissenschaft

Berechnen von Dingen gab es schon vor Computern. Im letzten Jahrzehnt gibt es einen Übergang zu Informationsprozessen. Bisher dachten wir, Informatik beschreibt künstliche dinge und nicht die reale Welt.

2.5.1 1940er Werkzeuge

Berechnungen heben andere Gebiete auf neue Ebenen: Verschlüsselungen knacken (Enigma), Atomenergie, Flugzeuge, Autos, Schiffe, Medikamente, Gebäude, Wettervorhersage.

2.5.2 1980er Methoden um Probleme zu Lösen

Nicht mehr wegzudenken aus vielen Feldern. Neues Wissen erschließen. Medikamente, neue Werkstoffe (Methan-resistentes Schild für Jupitersonde durch Ausrechnen aus Schrödinger-Gleichung)

2.5.3 2000- Prozesse

Biologie: Leseköpfe auf DNS Quantendynamik: Informationsübertragung, Berechnungen parallel Wirtschaft: Systeme auf Informationsflüsse untersuchen, Workflow

Teil der alltäglichen Kultur (im Gegensatz zu anderen Disziplinen sehr sichtbar) Web, iPod, eBay, Google, Wikipedia, Playstation, Xbox, Wii

Sieben Kategorien:

- Berechnung. Bedeutung und Grenzen von Berechnungen
- Kommunikation. Verlässliche Datenübertragung
- Koordination. Kooperation zwischen vernetzen Einheiten
- Speicherung. Speichern und Abrufen von Information
- Automatisierung. Bedeutung und Grenzen von Automatisierbarkeit
- Evaluation. Voraussage der Performanz und Kapazitätsplanung
- Design. Bauen von verlässlichen Software Systemen

3 Algorithmen

Nachdem wir einen Überblick über die Felder der Informatik gewonnen haben lernen wir nun die Grundbausteine der angewandten Informatik kennen, die Algorithmen. Wir führen als erstes die wichtigsten Sprachelemente ein, sehen uns dann an wie man die Geschwindigkeit von Programmen abschätzen kann und untersuchen dann verschiedene Verfahren zur Suche in Datenmengen.

3.1 Was ist ein Algorithmus?

Ein Algorithmus ist eine Art „Kochrezept“, welcher die Lösung eines Problems detailiert beschreibt. Die Beschreibung muss, damit der Computer sie umsetzen kann exakt sein. Das heißt alle Schritte und die Reihenfolge müssen unmissverständlich sein. Außerdem muss die Anzahl der durchzuführenden Schritte endlich sein.

3.2 Datentypen

Damit ein Algorithmus auch etwas tun kann benötigt er in der Regel Daten auf denen er arbeitet. Dabei unterscheidet man verschiedene Datentypen von denen die Zahlen schon aus der Mathematik bekannt sind.

Datentyp	Wird verwendet für
int	Ganze Zahlen
float	Fließkommazahlen
boolean	Wahrheitswert
string	Text

Tabelle 1: Datentypen (nicht vollständig)

Auf diesen Datentypen gibt es verschiedene Rechenmöglichkeiten. Zahlen lassen sich addieren, subtrahieren, multiplizieren und so weiter, Texte können aneinander gehängt werden oder es kann aus ihnen ein Teil ausgeschnitten werden. Mit Wahrheitswerten lassen sich Entscheidungen treffen.

3.3 Kontrollstrukturen

Um einen Algorithmus zu beschreiben genügen Daten natürlich noch nicht, es kommt auf die Folge der einzelnen Schritte an. Dazu benutzt man Kontrollstrukturen, die den Ablauf steuern.

In einem Algorithmus kommen zwei dinge immer wieder vor: Verzweigungen und Schleifen. Verzweigungen funktionieren wie Fallunterscheidungen. Falls zum Beispiel jemand ein Konto eröffnen will muss die Einverständniserklärung der Erziehungsberechtigten vorliegen, falls die Person minderjährig ist.

if *alter* < 18 **then**
  | Einverständnis holen
else
  | selber unterschreiben
end

Schleifen nutzen wir, wenn ein oder mehrere Schritte mehrfach ausgeführt werden müssen. Die einfachste Schleife tut eine Aufgabe für eine feste Anzahl von Wiederholungen.

for *i = 1..10* **do**
| gib i aus;
end

Als Variante benötigt man oft eine Schleife, die so oft wiederholt wird, bis eine bestimmte Bedingung eintritt.

stockwerk = 1;
while *stockwerk < 5* **do**
| fahre ein Stockwerk weiter nach oben;
end

3.4 Komplexität und O-Notation

Will man Algorithmen, die die gleiche Aufgabe erfüllen miteinander bezüglich des Aufwands vergleichen, so kann man die Komplexität bestimmen. Dazu betrachtet man wie viele Schritte beide für die gleiche Aufgabe benötigen wenn man die Aufgabe sehr oft wiederholt.

Wir können die obere Schranke an Wiederholungen bestimmen und mit der O-Notation aufschreiben. Die O-Notation beschreibt immer die obere Schranke, man kann auch die untere Schranke oder das Mittel an Schritten bestimmen. Betrachten wir dies an einem Beispiel.

Die Zahl n sei immer ganzzahlig durch 2 teilbar. Sie ist also von der Form $n = 2^m$. Wie viele Schritte sind nötig, bis n = 1 ist?

Input: n
while *n > 1* **do**
| teile n durch 2
end

Die Anzahl an Wiederholungen[1] beträgt $m = \log_2(n)$. Dies schreiben wir als O(log n).

Es gibt folgende Arten wie lange ein Algorithmus in der Ausführung benötigt:

O(1)	egal wie groß n wird, der Algorithmus braucht immer gleich lang
O(log n)	wächst ungefähr um einen konstanten Betrag, wenn sich das Argument verdoppelt
O($\sqrt{n}$)	wächst ungefähr auf das Doppelte, wenn sich das Argument vervierfacht
O(n)	wächst gleich wie n
O(n log n)	
O(n^2)	wächst quadratisch
O(2^n)	wächst exponentiell
O(n!)	wächst faktoriell

[1] Da der $\log_2$ in der Regel nicht auf dem Taschenrechner zu finden ist, kann man die folgende Formel benutzen:$\log_2(n) = \frac{\log_{10}(n)}{\log_{10} 2}$

Tabelle 2: Laufzeitverhalten von Algorithmen

Aufgabe: Potenzieren von 2^n

Schreibe einen einfachen Algorithmus auf mit dem man 2^n direkt berechnen kann. Multipliziere dazu 2 n-mal mit sich selbst. Berechne außerdem die Komplexität.

Um den Rechenaufwand zu vermindern gibt es einen kleinen Trick: Man kann 2^n auch als $\left(2^2\right)^{\frac{n}{2}}$ schreiben. Dadurch muss man deutlich weniger Rechenschritte durchführen. Schreibe auch hier einen Algorithmus zur Berechnung auf. Im schlechtesten Fall sind $2 \cdot log_2(n)$ Multiplikationen nötig. Vergleiche dies mit dem direkten Weg.

4 Suchen

Um unser Wissen über Algorithmen und Aufwandsabschätzungen zu vertiefen beschäftigen wir uns etwas mit dem Suchen. In Daten muss häufig gesucht werden. Typische Beispiele sind das Nachschlagen eines Worts im Wörterbuch, einer Adresse im Telefonbuch oder das Suchen nach einem Stichwort bei Google.

4.1 Lineare Suche

Bei der linearen Suche läuft man der Reihe nach durch eine Liste und vergleicht den aktuellen Eintrag mit dem Ausdruck nach dem man sucht. Sie funktioniert auch bei unsortierten Daten.

```
Input: suchwort, Liste
i = 0;
while i < Länge der Liste do
    if suchwort == Liste[i] then
        return i;
    end
    erhöhe i um 1;
end
```

Aufgaben

- Welchen Aufwand benötigt die lineare Suche im Mittel?
- Spiel: Zahlenraten im Bereich von 1 bis 100 mit linearer Suche und Zeit stoppen.
- Warum macht das Spiel in dieser Version wenig Spaß?

4.2 Binäre Suche

Die binäre Suche ist deutlich schneller als die lineare Suche. Sie basiert auf dem Intervallhalbierungsverfahren, funktioniert allerdings nur auf bereits sortierten Daten.

Aufgaben

- Welchen Aufwand benötigt die binäre Suche im Mittel?

```
Input: suchwort, Liste
rechts = Länge der Liste - 1;
if (suchwort < Liste[0]) oder (suchwort > Liste[rechts]) then
    return nicht gefunden;
end
links = 0;
mitte = 0;
while links <= rechts do
    mitte = (links + rechts) / 2;
    if Liste[mitte] == suchwort then
        return mitte;
    end
    if Liste[mitte] < suchwort then
        links = mitte + 1;
    else
        rechts = mitte - 1;
    end
end
return nicht gefunden;
```

- Spiel: Zahlenraten im Bereich von 1 bis 100 mit binärer Suche und Zeit stoppen.

4.3 Suche im Labyrinth

Schon im alten Griechenland verliefen sich immer wieder Menschen in Labyrinthen. Theseus kam dank Ariadne aus dem Labyrinth in Minos heraus, die im geraten hatte einen Faden zu spannen. Wenn keine Minotauren im Labyrinth sind kann man mit verschiedenen Algorithmen recht schnell den Ausgang finden.

Random Mouse

Folge einem Gang bis zu einer Kreuzung. Wähle per Zufall wo es weiter geht. Dabei sind Drehungen um 180 Grad verboten außer es geht nicht anders. Simuliert das Verhalten eines Menschen, der sich nicht erinnert wo er ist.

Sackgassen füllen

Fokus auf dem Labyrinth. Fülle alle Sackgassen bis zur nächsten Kreuzung. Dabei entstehen weitere Sackgassen bis die Lösung übrig bleibt.

Wand entlanglaufen

Folge einem Gang bis zur Kreuzung. biege nun immer rechts (links) ab.

Rekursives Backtracking

Falls an einer Wand gib Falsch zurück, am Ziel gib Fertig zurück, sonst versuche rekursiv in alle Richtungen zu laufen. Zeichne Linie auf und lösche sie falsch Falsch zurück kommt. Markiere besuchte Felder. Diese Suche wird auch *depth first* genannt.

Aufgaben

- Versuche die verschiedenen Algorithmen auf den ausgeteilten Blättern aus.

Noch mehr Material zu Labyrinthen gibt es unter:
`http://www.astrolog.org/labyrnth/algrithm.htm`

5 Kommunikation

Schon immer kommunizieren Menschen gern miteinander. Früher dauerte dies oft recht lange, als Briefe noch von Kutschen und Schiffen transportiert werden mussten. Seit dem Telegraphen war klar, dass es viel schneller gehen kann, mit der Erfindung des Telefons hat sich diese Entwicklung beschleunigt und heute können wir über das Internet viele Menschen innerhalb von Bruchteilen von Sekunden erreichen.

Ein sehr wichtiger Punkt ist dabei, dass die Übertragung unserer Stimme oder Texte, also der Daten, verlässich geschehen muss. Wie die Kommunikation über das Internet funktioniert wollen wir nun genauer untersuchen. Unser Anwendungsbeispiel dafür sei, dass wir von Wikipedia eine Webseite aufrufen wollen.

Zunächst ist die Frage zu beantworten was das Internet eigentlich ist. Es ist ein Netz von Netzen, d.h. es verbindet kleinere Netze zu einem großen Netz. Bevor wir uns mit dem Internet befassen müssen wir erst verstehen wie ein kleineres Netz funktioniert.

5.1 Lokale Netze

Schon seit vielen Jahren vernetzt man Computer lokal, also in einem Zimmer oder über ein Gebäude hinweg. Das ist sehr geschickt, denn man kann sich so teure Drucker oder Meßgeräte teilen und muss nur die Daten übertragen.

Damit Rechner untereinander vernetzt werden können müssen sie alle an ein Netzwerkkabel angeschlossen sein. Funkverbindungen wollen wir heute weglassen. Es gibt verschiedene Möglichkeiten um Rechner zu vernetzen. 2 PCs können direkt miteinander verbunden werden. Viele PCs können gemeinsam an einem Kabel sein oder einen Stern bilden.

Diese verschiedenen Vernetzungsarten (Topologien) haben auch Auswirkungen auf die Geschwindigkeit mit der Daten übertragen werden können. Denn weil die Übertragung digital erfolgt kann immer nur ein Rechner „sprechen“. Immer wenn mehr als einer Daten sendet gibt es eine Kollision und die Daten beider Rechner sind unbrauchbar. Darum bedient man sich einiger Tricks um möglichst wenige Kollisionnen zu haben. Jeder Rechner sendet nur Pakete einer bestimmten Länge und alle Rechner hören mit, ob gerade einer sendet. Bei einer Kollision hören alle Sender auf und senden danach zufällig neu, so gibt es weniger Kollisionen.

Größere Mengen an Daten müssen also in viele kleine Pakete zerlegt werden. Diese werden nummeriert, damit sie auf der Gegenseite in der richtigen Reihenfolge zusammengesetzt werden können.

Damit Daten übertragen werden können brauchen alle Rechner eindeutige Namen. Zum Glück hat jede Netzwerkkarte einen Chip, der eine eindeutige Nummer enthält, die MAC-Adresse (Media Access Controll). Leider ist diese Adresse für Menschen unhandlich.

Dehalb vergibt man an Rechner lieber IP-Adressen, d.h. man verwendet das selbe Adressschema wie im Internet auch lokal. Die Adressen bestehen immer aus vier Zahlen zwischen 0 und 255, die durch Punkte getrennt werden. Für lokale Netze nimmt man Adressen die mit `192.168.` beginnen.

Für jeden Rechner wird auch noch eine Subnetz-Maske eingestellt, diese sagt aus welcher Bereich der Adresse lokal ist. Eine Maske von `255.255.255.0` bedeutet, dass nur die letze Zahl sich ändert und bis zu 256 Rechner im Subnetz sind.

Um das eigene Subnetz zu verlassen stellt man einen Gateway ein. Dies ist je nach Topologie ein Rechner irgendwo entlang eines Kabels oder der zentrale Rechner in einem Stern. Vom Gateway aus kann man dann andere Netze erreichen. Genauer werden wir das im nächsten Teil sehen.

Damit man nicht jedem Rechner von Hand alle diese Zahlen eintippen muss gibt es DHCP, ein Verfahren bei dem automatisch ein Rechner allen anderen Rechnern in einem lokalen Netz ihre IP-Adressen, Subnetz-Masken und Gateways mitteilt.

Mit dem Befehl `ipconfig /all` oder `ifconfig` unter Linux kann man alle wichtigen Informationen über das Netzwerk des eigenen Rechners herausfinden.

Aufgaben

- Finde mit `ipconfig /all` die MAC-Adresse deines Rechners heraus.
- Finde die IP-Adresse deines Rechners heraus.

5.2 Internet

Das Internet verbindet viele lokale Netze zu einem großen Netz. Es ist hierarchisch aufgebaut und über die Adressen ist einfach zu bestimmen in welchem Teil des Netzes welche Rechner erreichbar sind. Diese Adressierung nennt man IP4 (Internet Protocoll version 4). Wenn man kein lokales Netz hat, so sagt wie IP-Adresse einiges über den Standort des Rechners aus.

An der Universität Ulm hat z.B. ein Rechner die IP-Adresse 134.670.208.100. Um den Server der New York Times mit der Adresse 199.239.136.245 zu kontaktieren verbindet sich der Rechner erst zu seinem Gateway, von dort in das höhere Subnetz 134.60.70, dann weiter auf die Ebene der Universität (134.60) und dann auf die Ebene von Deutschland um von dort in die USA, und dort über mehrere weitere Stellen an den Rechner der Zeitung. So können nun beide Rechner miteinander kommunizieren.

Mit dem Befehl `ping` können wir ein Datenpaket an einen anderen Rechner senden um zu sehen, ob er eingeschaltet ist. Bei einem Ping wird eine besondere Art von Paket gesendet, die Steuerinformationen enthält. Wenn wir wissen möchten auf welchem Weg das Paket reist, so können wir den Befehl `tracert` benutzen (`tracepath` unter Linux), der uns entlang jeder Station eine kurze Information zurück schickt. Im Grunde ist dieser Befehl nur ein mächtigerer Ping.

Bisher haben wir nur IP-Adressen verwendet, wir nutzen aber sonst immer Namen als Adressen weil diese für Menschen leichter zu merken sind. Es gibt darum eine Art Telefonbuch, das die Namen auf IP-Adressen abbildet. Dieses nennt man DNS (Domain Name System).

Mit dem Befehl `nslookup` kann man den Namen eines Rechners in eine IP-Adresse auflösen. Die Namensauflösung geschieht auch hierarchisch, die Namen der lokalen Rechner sind einem Name-Server lokal bekannt, für Namen die nicht lokal sind muss ein Server höher in der IP-Hierarchie gefragt werden, der z.B. alle größeren Name-Server auf der Ebene `de` kennt.

Aufgaben

- Pinge den Rechner deiner Nachbarsgruppe.
- Finde heraus welche IP-Adresse `www.uni-ulm.de` hat.
- Verfolge über traceroute wie du `www.uni-ulm.de` erreichst.

- Vergleiche den Weg mit mindestens einer anderen Gruppe. Wieviel ist gleich?
- Finde heraus welche IP-Adresse `vimp.de.vu` hat.
- Verfolge über traceroute wie du `vimp.de.vu` erreichst.
- Vergleiche den Weg mit mindestens einer anderen Gruppe. Wieviel ist gleich?

5.3 Protokolle

Leider genügt es nicht, dass wir nun Verbindungen von einem Rechner zu einem anderen aufbauen können. Der Zielrechner muss auch noch wissen was wir von ihm wollen, ob wir eine Webseite abholen möchten, über chat mit jemand sprechen wollen, per Internet-Telefonie mit jemand telefonieren oder vielleicht eine Videokonferenz halten möchten.

Weil es also viele verschiedene Dienste gibt muss es einen Mechanismus geben um sie auseinander zu halten. Dazu wurden Ports erfunden. Ein Port ist eine Zahl, die angibt welchen Dienst auf dem anderen Rechner wir in Anspruch nehmen möchten.

Port	Name	Beschreibung
20	ftp	Dateien übertragen
22	ssh	Sicheres Anmelden auf Rechner und dort Befehle ausführen
23	telnet	Anmelden auf Rechner und dort Befehle ausführen
25	smtp	E-Mail versenden
53	dns	Namen von Rechnern nachschlagen
80	http	Webseiten anschauen
110	pop3	E-Mail lesen
4662	EMule	Filesharing
6969	Bittorrent	Neueres Filesharing-Protokoll

Tabelle 3: Bekannte Ports und ihre Namen

Der Port wird immer mitübertragen, selbst wenn wir ihn nicht direkt angeben müssen, wenn wir z.B. im Internet surfen.

Zu jedem Port gibt es nun noch ein eigenes Protokoll. Denn das Verfahren um eine Webseite anzusehen ist natürlich anders als das um sich entfernt auf einem Rechner anzumelden und dort Befehle auszuführen.

Wir sehen uns jetzt kurz an, wie http (Hypertext Transfer Protocol) funktioniert, mit dem auf Webseiten zugegriffen wird. Im Grund ganz einfach, eine Anfrage nach einer bestimmten Seite wird mit dem Befehl `GET` gestellt und danach liefert der Webserver aus in welcher Form die Seite kommt und danach die Seite selbst.

```
GET / HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 12 Jan 2009 10:18:26 GMT
Server: Apache/2.2.3 (Debian) mod_jk/1.2.18
PHP/5.2.0-8+etch13 mod_ssl/2.2.3 OpenSSL/0.9.8c
Last-Modified: Fri, 23 Nov 2007 14:52:29 GMT
ETag: "a52be-e44-c37ae540"
Accept-Ranges: bytes
Content-Length: 3652
```

```
Connection: close
Content-Type: text/html

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
  Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
...
```

Mit dem Befehl `telnet` ist es möglich von Hand solche Befehle abzuschicken. Dazu muss man aber den Port mit angeben. Der Befehl sieht dann so aus:
`telnet de.wikipedia.org 80`

Anschließend gibt einen Befehl ein. Der Befehl muss im folgenden Format sein:
`GET Pfad/Seite.html http/1.1` also zum Beispiel
`GET /wiki/Informatik http/1.1` um die Seite `http://de.wikipedia.org/wiki/Informatik` zu bekommen.

Dabei bedeutet `GET`, dass man eine Seite von dem Server haben möchte. `http/1.1` ist die Version des Protokolls (und darf weggelassen werden) und davor geben wir an welche Seite wir möchten.

Der Server antwortet daraufhin mit einem Statuscode `200 OK` und liefert danach einige Informationen über sich und die Seite, bevor die Seite selbst kommt.

Aufgaben

Jetzt können wir endlich die Webseite vom Anfang abfragen! Schau dir mit `telnet` verschiedene Seiten von Wikipedia an, zum Beispiel die Informatik-Seite.

Zum Schluss

Was ist die wichtigste Sache, die du heute gelernt hast?

Weiterführende Links

`http://de.wikipedia.org/wiki/Ethernet`
Lokale Netze werden heute so gut wie ausschließlich mit Ethernet realisiert.

`http://de.wikipedia.org/wiki/Address_Resolution_Protocol`
MAC-Adressen werden durch dieses Protokoll aufgelöst.

`http://de.wikipedia.org/wiki/IPv4`
IP-Adressen und was alles dazugehört.

`http://de.wikipedia.org/wiki/Hypertext_Transfer_Protocol`
Und schließlich http, damit man auch mal eine Webseite ansehen kann.

`http://de.wikipedia.org/wiki/HTTP-Statuscode`
Codes die der Server antwortet.

6 Datenschutz und Privatsphäre

Im heutigen Termin wollen wir uns mit Datenschutz und der Privatsphäre beschäftigen.

6.1 Privatsphäre

Hier sind drei kurze Punkte als Grundlage zur Privatsphäre:

Man sagt, dass das Netz nichts vergisst. Hast du schon Erfahrungen mit dem Netz gemacht, in dem dies ein Vor- oder ein Nachteil war?

Viele Millionen Menschen sind heute in online Communities wie Facebook, StudiVz usw. registriert. Diese Gemeinschaften haben Vor- und Nachteile, man kann viel über andere Nutzer erfahren, Leute können sich als jemand anders ausgeben, man kann neue und alte Freunde (wieder-) finden.

- Das Netz vergisst nichts: Finde so viel wie möglich über das Internet über Michael und Guido heraus wie möglich.
- Online-Communities: Sammle drei Vor- und drei Nachteile von solchen Gemeinschaften. Was kann man tun um keine Nachteile zu erfahren?

6.2 Datenschutz

Als Datenschutz wird der Schutz personenbezogener Daten vor Missbrauch bezeichnet. In Deutschland gibt es Datenschutz als Gesetz bereits seit 1970. 1983 wurde in Zusammenhang mit dem Volkszählungsurteil des Bundesverfassungsgerichts der Begriff des informationellen Selbstbestimmungsrechts eingeführt. Das ist das Recht des Einzelnen, grundsätzlich selbst über die Preisgabe und Verwendung seiner persönlichen Daten zu bestimmen.

Hauptprinzipien des Datenschutzes sind

- Datensparsamkeit und Datenvermeidung,
- Erforderlichkeit,
- Zweckbindung.

Falls Daten einmal angefallen sind müssen diese durch technische und organisatorische Maßnahmen geschützt werden. Insbesondere muss sicher gestellt werden, dass nur berechtigte Personen Zugriff auf die Daten erhalten.

Werden die Daten nicht mehr benötigt müssen sie gesperrt werden. Unter Umständen müssen die Daten erst noch eine Zeit aufbewahrt werden, z.B. für die Steuer 10 Jahre. Nach Ablauf der Fristen sind die Daten dann zu löschen.

Seit Anfang dieses Jahres sollen alle Verbindungen die über das Internet geschehen gespeichert werden (Vorratsdatenspeicherung). Datenschützer glauben, dass sich dies auf das tägliche Leben und Verhalten der Menschen auswirken wird.

Aufgaben

- „Datenschutz ist Täterschutz“ – argumentiere in einem Satz für oder gegen diese Aussage.
- Kennst du Fälle in denen unverantwortlich mit Daten umgegangen wurde?

- Vorratsdatenspeicherung: Wie glaubst du wirkt sich die Vorratsdatenspeicherung auf dein Leben aus? Wie auf andere Menschen? Schreibe je einen oder zwei Sätze dazu.

6.3 Plenumsdiskussion

Wir tragen die Ergebnisse der Aufgaben zusammen und sprechen dann noch je nach Wunsch über folgende Themen:

- Was ist an der Aussage „Ich habe doch nichts zu verbergen“ dran?
- Privatsphäre: Wie kann ich mich schützen? (Datensparsamkeit und Datenvermeidung, Pseudonymisierung, Anonymisierung)

Zum Schluss

Was ist die wichtigste Sache, die du heute gelernt hast?

Weiterführende Links

`http://www.datenschutz.de/` Virtuelles Datenschutzbüro.

`http://www.foebud.org/` Verein zur Förderung des öffentlichen bewegten und unbewegten Datenverkehrs e. V.

`http://dejure.org/gesetze/StGB/238.html` Gesetz gegen Stalking § 238 StGB.

Es gibt in Deutschland noch kein Gesetz gegen Mobbing, wohl aber gegen üble Nachreden (§ 186 StGB), Beleidigungen (§ 185 StGB), sowie Gesetze zu Abmahnungen, Kündigungen usw.

7 Sicherheit

Technische Systeme lassen sich angreifen. Es gibt genügend Interessen Systeme zu manipulieren um sich Vorteile oder anderen Nachteile zu verschaffen. Manchmal auch nur „weil man's kann".

Computer werden zum Beispiel durch Viren, Würmer, Trojaner und Denial-of-Service Attaken angegriffen.

Doch nicht nur Computer können angegriffen werden. Verschiedene Regierungen spionieren ihre Bevölkerung aus oder haben es vor.

7.1 Hacker, Cracker und Systeme

Nun müssen wir uns noch etwas klarer werden, was und wen es alles in unseren Szenarien mit Angriffen auf Rechner usw. geben kann.

Als **Hacker** bezeichneten sich in den 80ern Leute, die in Systeme eindrangen um zu beweisen, dass diese nicht sicher sind. Sie legten großen Wert darauf keinen Schaden zu verursachen. Was heute in den Medien Hacker genannt wird nannte diese Gruppe **Cracker**, da diese Systeme knacken. Cracking wird mit dem Zerstören von wenigstens einem Teil des Systems verbunden.

Ein **Gut** (von *Güter*) wollen wir als Name für die Systeme, Gegenstände, Daten oder Personen verwenden, die wir schützen wollen. Ein System kann dabei z.B. ein PC oder Laptop sein, eine Software die über mehrere Rechner verteilt ist, eine Telefonverbindung, je nachdem. Ein Gut ist dabei noch weiter gefasst.

Ein paar Fragen

- Wurde dein Rechner schon einmal von Viren befallen?
- Gab es Fälle in deinem Bekanntenkreis in denen Rechner befallen waren?
- Was für Güter sollen wir nachher behandeln?

Eine **Bedrohung** ist ein System oder eine Person, die einen **Angriff** auf ein Gut ausüben kann. Mit einer gewissen **Wahrscheinlichkeit** findet ein Angriff dann auch in der realen Welt statt. Diese Wahrscheinlichkeit ist unser **Risiko**.

7.2 Was ist Sicherheit

Es gibt keine absolute Sicherheit. Vernünftig betrachtet geht es immer um eine Abwägung zwischen Aufwand und Risiken. Bruce Schneier stellt in „Beyond Fear" einen fünfstufigen Ansatz zur Beurteilung von Lösungen vor:

1. Welches Gut soll abgesichert werden?
2. Was für Bedrohungen gibt es?
3. Wie gut deckt eine Sicherheitslösung das Risiko ab?
4. Welche neuen Risiken erzeugt die Lösung?
5. Was für Kosten und Einschränkungen bringt die Lösung mit sich?

Wir werden dieses Schema später einsetzen um Lösungen zu beurteilen.

7.3 Arten von Schwachstellen

Bei jedem System gibt es Schwachstellen, bei Rechnern bzw. Software gibt es eine Reihe von bekannten und oft auftretenden Stellen. Eine kleine Auswahl:

- Verletzung des Speichers wie:
  - Buffer overflows
  - Dangling pointers
- Fehler bei der Überprüfung von Eingaben wie:
  - Format string bugs
  - Falsche Behandlung von shell metacharacters (Interpretieren)
  - SQL injection
  - Code injection
  - E-mail injection
  - Directory traversal
  - Cross-site scripting
- Race conditions
- Privilege-confusion bugs, wie:
  - Cross-site request Fälschungen
  - Clickjacking
  - FTP bounce attack
- User interface Versagen wie:
  - Warning fatigue oder user conditioning
  - Blaming the Victim (Den Benutzer etwas entscheiden lassen ohne ihm die nötigen Informationen zu geben)

7.4 Häufige Bedrohungen

Je nach Gut gibt eine Reihe von Risiken, hier sollen exemplarisch ein paar davon mit Lösungsansätzen vorgestellt werden.

Diebstahl

Gerade mobile Geräte können gestohlen werden oder verloren gehen. Mobiltelefone bieten lassen sich nur schlecht sichern, Laptops können mit Kabelschlössern festgemacht werden, die Daten auf ihnen lassen sich verschlüsseln und nur über Passwort freigeben. So ist im schlimmsten Fall nur das Gerät fort, keine vertraulichen Daten.

Kosten: Einrichten der entsprechenden Programme, einarbeiten in theoretischen Hintergrund.

Phishing-Mails

Während Spam-Mails fast immer erkannt werden sind Phishing-Mails wie echte Mails von der Bank oder einem Geschäftspartner gemacht. Sie bitten immer wieder um geheime Daten, z.B. die Eingabe der PIN auf einer unsicheren Seite.

Kosten: Ständige Kontrolle von eigehenden Nachrichten nötig. Grundsätzlich sollten Mails von Banken vom Empfänger sehr streng geprüft werden. Nie Links in den Mails folgen, falls die Mail suspekt ist.

Viren und Trojaner

Beides wird über Dateien verbreitet. Es sind viele verschiedene Viren im Umlauf und laufend werden neue erfunden. Antivirussoftware fängt die meiste Schadsoftware ab.

Kosten: Es gibt freie Antivirussoftware, lediglich die Arbeit für Installation und leichte Performanceeinbußen während Überprüfungsphasen sind hinzunehmen.

Alternativ kann man auf sicherere und/oder wenige verbreitete Betriebssysteme wie Linux oder MacOS ausweichen.

Sehr problematisch ist, dass die Bundesregierung Trojanersoftware einsetzen will um Bürger auszuspähen.

7.5 Wie kann ich mein System schützen?

Natürlich kann hier nicht jedes Risiko behandelt werden, aber eine kleine Auswahl von häufigen Risiken soll schon gegeben werden.

An erster Stelle sollte die physikalische Sicherheit des Guts stehen. Ein Gerät kann meistens mit wenig Aufwand vor Diebstahl geschützt werden. In den meisten Fällen wird dieses Risiko gering sein (Rechner daheim).

Um auf die Daten wieder zugreifen zu können empfiehlt es sich Backups zu machen. Dies hilft auch im Fall von defekter Hardware.

Um Daten vor anderen zu schützen können diese verschlüsselt werden. Sollen sie an andere übermittelt werden, so muss vorher sichergestellt werden, dass nur der Empfänger die Daten entschlüsseln kann.

Je nach Risiko kann es sinnvoll sein auch die Kommunikation zu verschlüsseln. Dank unverschlüsselter E-Mails und anderer Kommunikation ist Wirtschaftsspionage ein großes Problem.

Daten die man von anderen erhält sind zuerst immer zu prüfen ob sie mit Viren oder ähnlichem befallen sind.

Wenn man Windows einsetzt sollten diverse Systemdienste deaktiviert werden, Updates sind regelmäßig durchzuführen! Es kann sich lohnen auf ein anderes Betriebssystem zu wechseln.

Schließlich sollte man eigene Daten nur sparsam preisgeben.

Zum Schluss

Was ist die wichtigste Sache, die du heute gelernt hast?

Weiterführendes

Bruce Schneier: *Beyond Fear: thinking sensibly about security in an uncertain world*. Copernicus Books, 2003. ISBN 0-387-02620-7

`http://www.cert.org/` Carnegie Mellon University's Computer Emergency Response Team (CERT).

`http://www.cert.dfn.de/` CERT des Deutschen Forschungsnetzes.

8 Urheberrecht

Zum heutigen Termin wollen wir uns mit dem Urheberrecht beschäftigen. Vieles hat sich geändert oder ist dabei anders zu werden. Insbesondere durch die Digitalisierung.

Zuerst soll uns aber die Frage beschäftigen, was man mit einem Gegenstand alles machen kann.

- Verkaufen
- Verleihen
- Verschenken
- Drauf Schlafen
- Den Hund dran kauen lassen

Ein Teil dieser Eigenschaften wird vom Urheberrecht und anderen Gesetzen geregelt (verkaufen, verschenken), ein Teil fällt unter Fair Use, d.h. das Gesetz erlaubt uns die Sachen weiter zu verwenden (verleihen) und ein Teil wird nicht von Gesetzen geregelt. Damit ergibt sich das folgende Bild in dem der Bereich Fair Use grau ist.

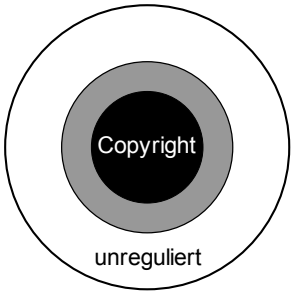


Abbildung 1: Bereiche des Urheberrechts: Geregelt, Fair Use und unreguliert.

In dem Moment, in dem wir es aber nicht mehr mit Gegenständen sondern Daten zu tun haben ändern sich viele dieser Punkte.

- Verkaufen (und wer löscht das Original?)
- Verleihen (wird die Kopie gelöscht?)
- Verschenken (ok)
- Drauf Schlafen
- Den Hund dran kauen lassen (geht nicht mehr)

Einige Möglichkeiten werden nicht mehr gehen, denn es handelt sich ja nun um Daten. Bei den anderen Punkten gibt es zwar Gesetze, aber die Umsetzung ist nicht mehr so einfach wie mit einem Gegenstand.

8.1 Regulierung

Um genauer zu verstehen, welche Faktoren auf diese Punkte wirken ist es nötig einmal das Umfeld anzusehen. Vier verschiedene Faktoren wirken sich darauf aus, was wir mit Daten alles machen können. Dies sind Normen, Gesetze, der Markt und die Architektur.

In Abbildung 2 kann man die vier Faktoren sehen. In der Mitte sind die zu regulierenden Daten. Je nachdem welcher Bereich stärker ist, kann sich eine Verschiebung ergeben.

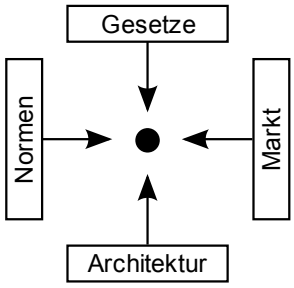


Abbildung 2: Faktoren bei der Regulierung.

8.1.1 Normen

Normen sind ethische Regeln, die sich durch die Gemeinschaft mit anderen Menschen herausgebildet haben. Typische Normen beantworten die folgenden Fragen: Was darf ich? Was darf ich nicht? Was ist *richtig*?

8.1.2 Der Markt

Der Markt bietet Waren an und hat ein Interesse daran, diese, den Verkaufsweg und ähnliches zu schützen. Dabei streben alle Firmen Monopole an, falls möglich.

Der Markt bietet aber auch zusätzlich die Möglichkeit z.B. gebrauchte Waren erneut in Umlauf zu bringen.

8.1.3 Gesetze

Gesetze regeln viele Dinge wie den Kauf und Verkauf von Waren. Sie können sich im Lauf der Zeit ändern und müssen immer wieder an die aktuelle Lage angepasst werden. Optimal sind Gesetze, die nicht auf Technologien festgelegt sind.

8.1.4 Architektur

In diesem Bereich wirken Kopierschutzverfahren wie CSS oder DRM (Digital Rights Management). DRM kann z.B. sehr feingranular regeln was man alles mit einem PDF-Dokument tun darf. Diese Regelungen können sich direkt auf den Fair Use Bereich auswirken.

Aktuelles Beispiel: Kindle soll Bücher nicht vorlesen dürfen. Was machen dann Blinde?

8.2 Ausblick

Nach unserer kurzen Betrachtung inklusive Diskussion stellt sich die Frage, was sinnvoll wäre. Die aktuellen Regelungen beschränken viele Benutzer und zu einem guten Teil unnötig. Vielen Firmen und Verbänden merkt man an, dass sie die

Möglichkeiten und neuen Verhaltensweisen der Zeit der digitalen Daten noch nicht verstanden haben.

Aufgaben

Schreibt für alle vier Faktoren auf:

- was ihr ändern würdet. Konzentriert euch dabei auf maximal drei wichtige Aspekte.
- wie man die Änderungen umsetzen könnte.
- Sucht im Internet, ob es euren Ansatz schon gibt :-)

8.3 Raum für Lösungen

Es gibt eine Reihe von Lösungsmöglichkeiten. Verschiedene neuere Lizenzmodelle sind darauf ausgelegt die Probleme der aktuellen Regelungen zu reduzieren.

Free Document License

Die FDL ist eine Lizenz die

Creative Commons

Die Creative Commons Lizenz erlaubt verschiedene Abstufungen. Es ist möglich für private Zwecke mehr Rechte zu geben als für den kommerziellen Einsatz, Veränderungen zu erlauben oder auszuschließen und vieles mehr.

Änderungen am bestehenden Recht

Eine weitere Möglichkeit wäre, durch Einfluß auf die Politik das Urheberrecht zu einer praktischeren Variante zu ändern. Man könnte z.B. vorschreiben, dass man geschützte Werke zentral registrieren kann wie dies vor über 100 Jahren bereits der Fall war.

Eine solche Registrierung wäre verlängerbar, sollte aber insgesammt kürzer gelten als heutzutage.

Es wäre auch sehr nützlich, mehr Rechte für Transformationen, Collagen und ähnliches einzuräumen, da heute immer mehr solche abgeleitete Werke entstehen.

Zum Schluss

Was ist die wichtigste Sache, die du heute gelernt hast?

Weiterführendes

Lawrence Lessig: *Free Culture, The Nature and Future of Creativity*. Penguin Books, 2004. ISBN 0-14-303465-0

`http://www.creativecommons.org/` Creative Commons.

`http://www.gnu.org/copyleft/fdl.html` Lizenz für freie Dokumente.

`http://dejure.org/gesetze/UrhG` Urheberrecht, alle Gesetze online.

9 Kryptografie

10 Sprachen

HTML, CSS XML SVG imperative funktionale logische oo dfa, Regex ganz toll machbar

11 Kompression

Kompression ist notwendig um große Datenmengen handhabbar zu machen, so werden Filme oder Musik so klein, dass wir sie einfach mitnehmen können. Für einen unkomprimierten Film würde man sonst viele Gigabyte benötigen, für die Lieder die man auf einem mp3-Player mit sich trägt wären viele CDs nötig.

Man unterscheidet zwischen verlustfreier und verlustbehafteter Kompression. Für Texte bevorzugen wir verlustfreie Kompression, für Bilder und Filme genügen in der Regel auch die verlustbehaftete Varianten.

Einführung Verlustfrei RLE, LZW

11.1 Verlustfreie Kompression

Als Beispiel für verlustfreie Kompression betrachten wir die Lauflängenkodierung. Sie wird für Bilder und Texte verwendet, z.B. bei Faxen.

Bei der Lauflängenkodierung werden Folgen von gleichen Dingen zusammengefasst, das Verfahren eignet sich deshalb besonders gut für Daten mit viel Wiederholungen. Eingesetzt wird es z.B. bei Windows Bitmaps (BMP, RLE).

Um zu unterscheiden wo Wiederholungen sind müssen Marker gesetzt werden. Marker sind dabei Zeichen die nicht in den zu komprimierenden Daten vorkommen.

Beispiel

Es sollen keine Zahlen in unseren Daten vorkommen und wir möchten gern ein Fax komprimieren. Das Fax besteht aus schwarzen und weißen Bildpunkten (S und W). Dann wird aus

```
WWWWWWWWWWSWWWWWWWWWWSSS
```

nach der Komprimierung

```
10W1S10W3S
```

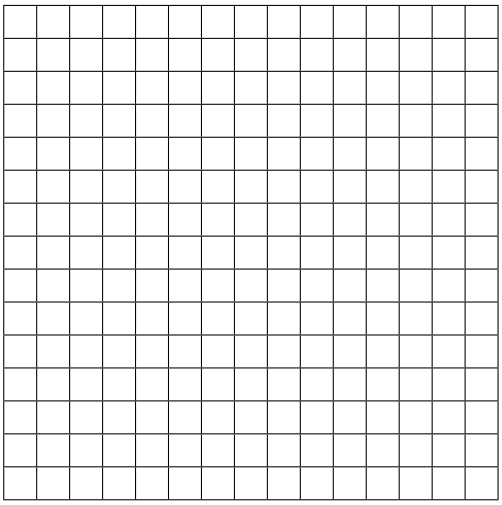
Aufgaben

Warum braucht ein Fax mit viel Text länger als ein Fax mit wenig Text?

Entschlüssele die folgende Sequenz und zeichne sie in die Matrix unten ein, dabei sollst du Felder mit W leer lassen und die mit S ausfüllen.

15w
6w3s6w
4w2s3w2s4w
3w1s7w1s3w
2w1s9w1s2w
2w1s2w1s3w1s2w1s2w
1w1s11w1s1w
1w1s11w1s1w
1w1s3w1s3w1s3w1s1w
2w1s3w3s3w1s2w
2w1s9w1s2w
3w1s7w1s3w
4w2s3w2s4w

6w3s6w
15w



LZW

Weiterführende Links

`http://de.wikipedia.org/wiki/Rle`
Details zu Lauflängenkodierung

12 Kompression 2

verlustbehaftet Audio Bilder Video

12.1 Audio

Heute nehmen viele Leute ihre Musik auf Mobiltelefonen oder kleinen Playern mit, früher wäre das schwer möglich gewesen, denn da musste man recht große Plattenspieler verwenden um überhaupt Musik hören zu können.

Aufgabe

Wie viel Daten braucht man für eine Minute Musik von CD (bei 44,1 kHz Samplingrate)? Um eine grobe Näherung zu bekommen kann man sich überlegen wieviel Daten auf eine CD passen.

Welche Verfahren kennst du bereits um die Datenmenge zu reduzieren?

12.2 Abtastung, Fouriertransformation

Um die folgenden Abschnitte zu verstehen benötigt man einen kurzen Einblick in das Abtasttheorem und Fouriertransformation.

Möchte man Musik, Bilder, usw. mit einem Computer, MP3-Player, anhören bzw. ansehen, müssen der Klang bzw. das Bild in digitaler Form gespeichert werden. Digital bedeutet, dass das Signal zeit- und wertdiskret ist, d.h. beispielsweise bei einem Musikstück, dass man das elektrische Signal, das ein Mikrofon liefert nur in bestimmten Zeitabständen misst. Die gemessene Spannung kann nur diskrete Werte annehmen. Die Natur ist im Gegensatz dazu analog. D.h. zeit- und wertkontinuierlich.

Man kann zeigen, dass man ein Signal, das mit der doppelten im Signal vorkommenden Frequenz abgetastet wird vollständig rekonstruiert werden kann. Tastet man mit einer niedrigeren Frequenz ab, gibt es einen Effekt, der sich Aliasing nennt.

Die Idee, die hinter der Fourier-Reihe steckt, ist, dass man ein Signal als Überlagerung von Sinus- und Kosinusfunktionen verschiedener Frequenz darstellen kann.

Der umgekehrte Weg, also ein Signal in seine Anteile verschiedener Sinus- und Kosinusfunktionen zu zerlegen nennt man Fourier-Transformation. Damit lässt sich das Frequenzspektrum einer Funktion berechnen, dass man bei der Kompression von Audio- und Bilddateien bearbeitet.

Hat man ein zeitdiskretes, periodisches Signal verwendet man die Diskrete Fouriertransformation (DFT).

Funktionsweise

Wir möchten uns jetzt kurz damit beschäftigen wie diese Verfahren funktionieren. Wir entwickeln dies gemeinsam an der Tafel.

12.3 JPEG

JPEG oder kurz JPG geht auf die *Joint Photographic Experts Group* zurück, die das Verfahren 1992 vorstellte. Mit JPEG lassen sich Bilder verlustbehaftet komprimieren.

- Farbraumumrechnung
- Tiefpassfilterung
- Blockbildung und diskrete Kosinustransformation
- Quantisierung
- Umsortierung und Differenzkodierung

Wir sehen uns nun ein Beispiel für Artefakte mit dem Rechner an.

12.4 MPEG-1

Es gibt verschiedene Verfahren um Videodaten zu komprimieren. Wir befassen uns kurz mit den Prinzipien von MPEG, dass auf JPEG basiert.

Aufgaben

Ein Video von DVD hat 768x576 Pixel wobei jedes Pixel aus 3 Byte für die Farben besteht. Wenn in einer Sekunde 25 Bilder gezeigt werden, wie viele Byte braucht dann ein Kinofilm von 1,5 Stunden?

Ein komprimiertes Video in einem PC Standardformat hat ca. 700 MB, wie groß ist der Kompressionsfaktor?

13 Automatisierung

Heute wollen wir uns mit einfachen Algorithmen beschäftigen. Unter einem Algorithmus versteht man allgemein eine genau definierte Handlungsvorschrift zur Lösung eines Problems oder einer bestimmten Art von Problemen in endlich vielen Schritten.

Das kann im täglichen Leben ein Kochrezept sein, aber auch Noten zu einem Lied, eine Gebrauchsanweisung oder die Anleitung für ein Formular das ausgefüllt werden muss.

In einem Algorithmus werden Probleme in kleine Schritte zerlegt, die nacheinander ausgeführt werden. Dies wollen wir heute spielerisch üben.