

# 706.088 INFORMATIK 1

TEILGEBIETE DER INFORMATIK

# WIEDERHOLUNG

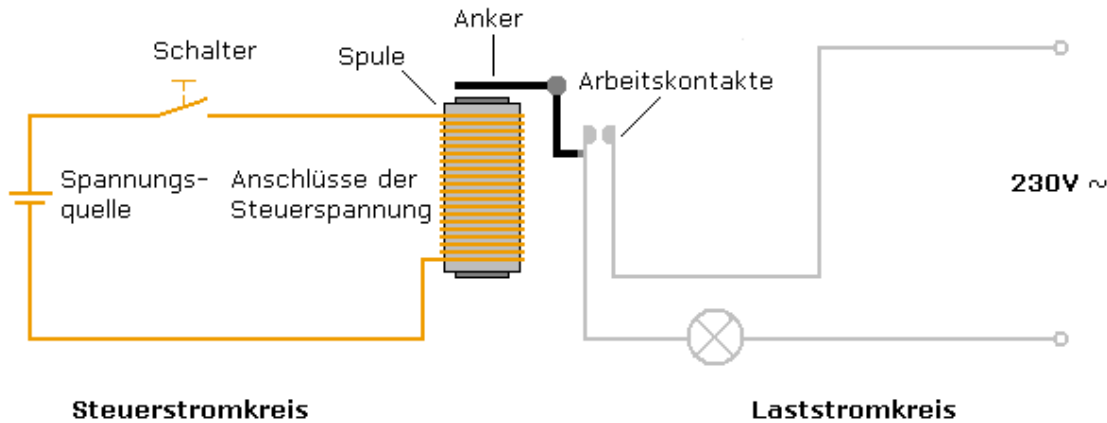
- › Funktionen
- › Geschichte:
  - ›› Mechanische Rechenmaschinen
  - ›› Elektronische Rechenmaschinen
- › Aufbau eines Computers
- › Moore's Law

# **RELAIS - ELEKTRONENRÖHRE - TRANSISTOR**

# RELAIS

- › Mechanischer Schalter
  - » Probleme:
    - › Umschalten dauert einige Sekundenbruchteile
    - › Benötigt viel Platz
    - › Taktfrequenz sehr beschränkt
    - › Mechanische Abnutzung

# RELAIS



Spule wird unter Strom  
gesetzt, Magnetfeld  
zieht Anker,  
Arbeitskontakte werden  
geschlossen, Strom  
kann fließen

Von [Stefan Riepl](https://commons.wikimedia.org/w/index.php?curid=10663175) in der [Wikipedia auf Deutsch](https://commons.wikimedia.org/w/index.php?curid=10663175) - Eigenes Werk, [CC BY-SA 2.0 de](https://commons.wikimedia.org/w/index.php?curid=10663175),  
<https://commons.wikimedia.org/w/index.php?curid=10663175>

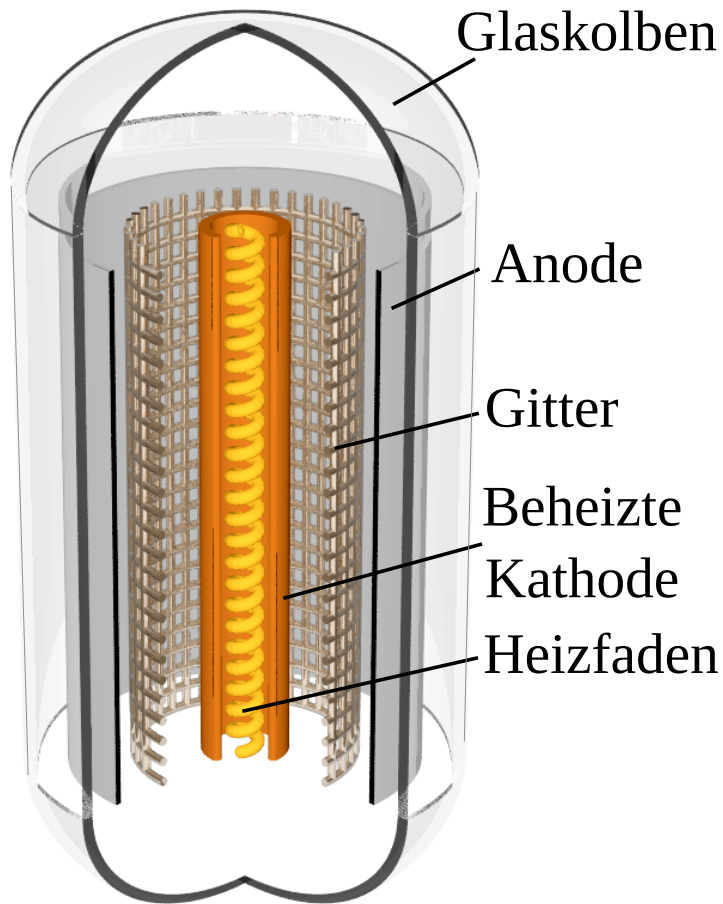
# ELEKTRONENROHRE



# ELEKTRONENRÖHRE

- › ist auch Schalter
- › 1000 mal schneller als Relais
- › Probleme:
  - » benötigt viel Strom
  - » Lebensdauer gering
  - » Programmierer für ENIAC waren eher Mechaniker

# ELEKTRONENRÖHRE



Stromführende Kathode,  
Stromaufnehmende Anode,  
Spannungsgefälle: Elektronen  
wandern von Kathode zu  
Anode, Strom fließt. Ist Gitter  
unter Strom werden  
Elektronen abgestoßen,  
Stromfluss stoppt.

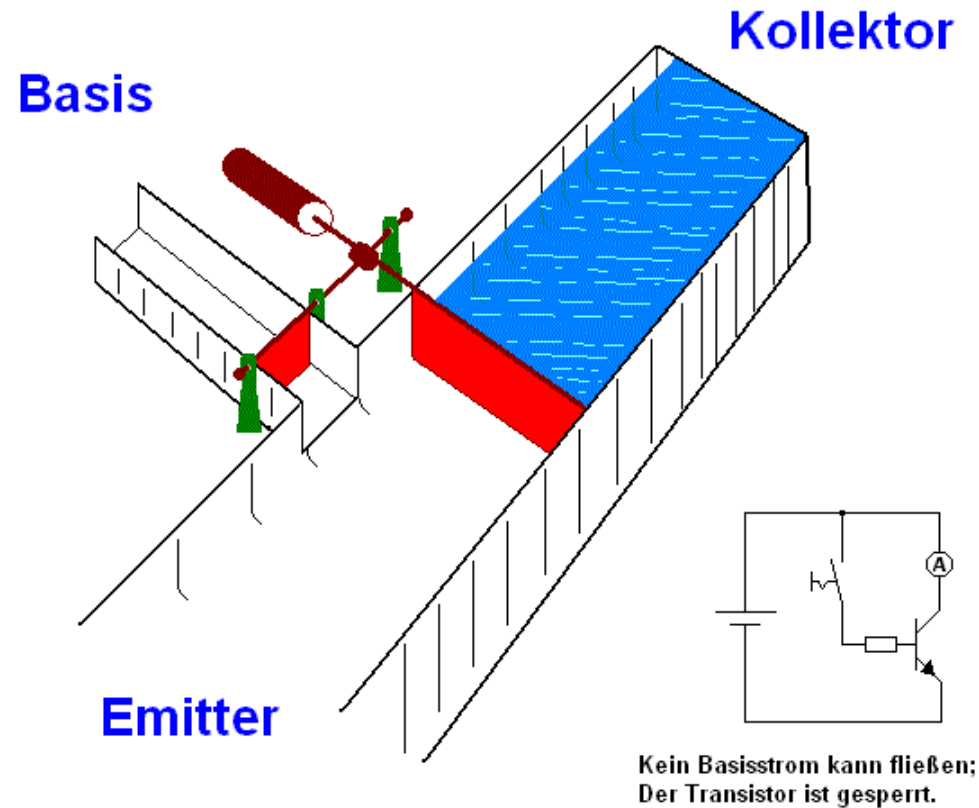
Von Svjo; German translation: Wdwd - [File:Triode-english-text.svg](https://commons.wikimedia.org/w/index.php?curid=50578519), CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=50578519>



# TRANSISTOR

- › Kleiner Strom zwischen **B** asis und **E** mitter schaltet großen Strom zwischen **C** ollektor und Emitter
- › Basis ist im Sperrbetrieb
  - » kein Strom fließt
  - » Kollektor wartet auf Strom
- › Wenn Spannung an Basis anliegt schaltet der Transistor
  - » Strom fließt zwischen Collector und Emitter
  - » Transistor leitet

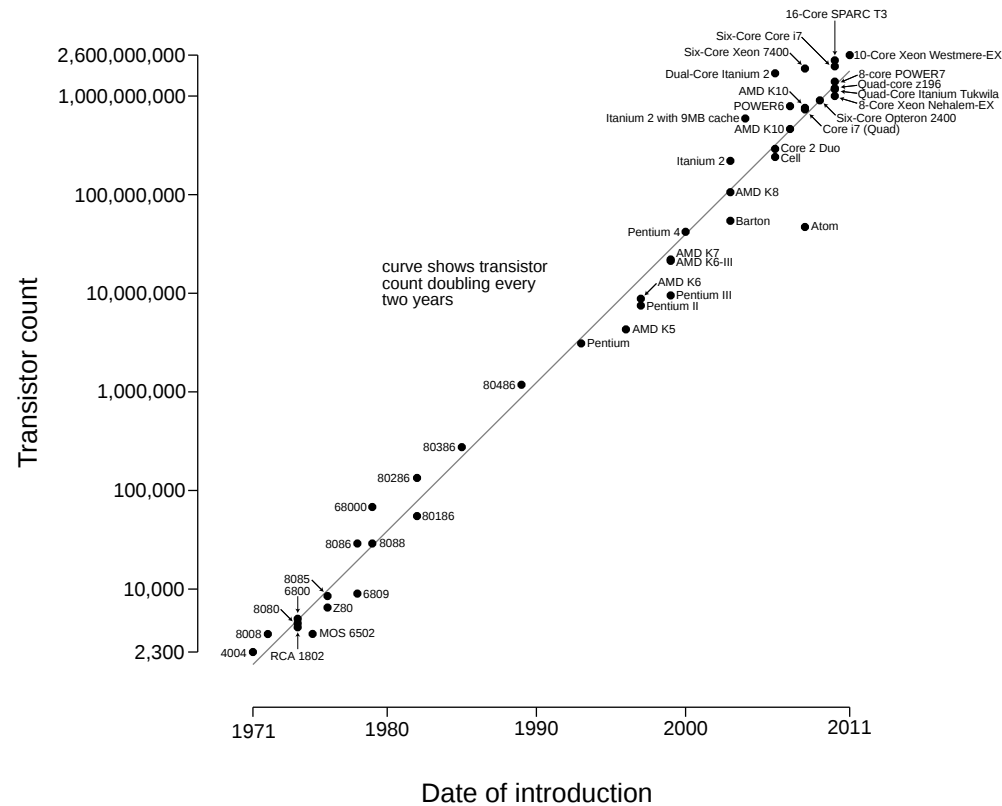
# TRANSISTOR



Von Stefan Riepl ([Quark48](#) 21:02, 2. Dez. 2007 (CET)) - Eigenes Werk (Originaltext: selbst erstellt), [CC BY-SA 2.0 de](#), [Link](#)

# MOORE'S LAW

## Microprocessor Transistor Counts 1971-2011 & Moore's Law



By Wgsimon - Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=15193542>

# TEILGEBIETE DER INFORMATIK

# TEILGEBIETE DER INFORMATIK

Die Informatik lässt sich in folgende Teilgebiete gliedern:

- › Theoretische Informatik
- › Technische Informatik
- › Praktische Informatik
- › Angewandte Informatik
- › Interdisziplinäre Informatik

# THEORETISCHE INFORMATIK

Beschäftigt sich mit (theoretischen) Grundlagenfragen der Informatik über formale Sprachen wie

- › Berechenbarkeitstheorie
- › Komplexitätstheorie
- › Logik
- › Graphentheorie
- › Kryptologie

# BERECHENBARKEITSTHEORIE

Prinzip: Welche Probleme sind mittels einer Maschine lösbar?

Ein Problem gilt als entscheidbar, wenn es durch einen Algorithmus gelöst werden kann.

› Beispiel:

›› Aussage der Prädikatenlogik erster Stufe

---

$$\forall x: x \leq y$$

---

› Alonzo Church und Alan Turing führten den Beweis, dass diese Beispiele nicht automatisch gelöst werden können.



# KOMPLEXITÄTSTHEORIE

Befasst sich mit der Klassifikation der Menge aller algorithmisch behandelbaren Problemen.

- › Laufzeit
- › Speicherbedarf

Verschiedene “Schwierigkeitsstufen”

- › Konstant, linear, quadratisch, polynomial

# KOMPLEXITÄTSTHEORIE

## P-NP Problem

- › P: Praktisch lösbar
- › NP: Praktisch (vermutlich) nicht lösbar
  - ›› Lösungen basieren auf **nichtdeterministischem** Modell
  - ›› Probleme in NP wachsen stärker als polynomiell mit ihrem Input (=NP-Vollständig)

# AUTOMATEN UND FORMALE SPRACHEN

Automaten stellen ein abstraktes Modell eines Computers dar

- › Verhalten sich gemäß bestimmter Regeln
- › Regeln sind in formalen Sprachen definiert
- › Werden verwendet um gewisse Eigenschaften von Algorithmen zu testen und zu beweisen.

# LOGIK

“Lehre des vernünftigen Schlussfolgerns”

- › **Formale Logik:** Untersucht Aussagen (nicht den Inhalt) auf ihre Gültigkeit.
- › **Aussagenlogik**
  - ›› Befasst sich mit Aussagen (Atomen mit Richtig/Falsch zuweisung) und deren Verknüpfung (Junktoren)
- › **Prädikatenlogik**
  - ›› Erlaubt die Darstellung der inneren Struktur von Sätzen über Prädikate (über Termen, Funktoren, Prädikatoren, Quantoren).

# KRYPTOLOGIE

Die Wissenschaft der Informationssicherheit.

- › Digitale Signaturen
- › Identifikationsprotokolle
- › Geheimnisteilung
- › **Symmetrische** Kryptosysteme
  - ›› DES, AES Verschlüsselungen
- › **Asymmetrische** Kryptosysteme
  - ›› PGP, RSA Verschlüsselungen
  - › Private Key, Public Key

# SYMMETRISCHE KRYPTOSYSTEME

ein Schlüssel für Ver- und Ent-Schlüsselung

› Beispiele:

» ROT13, 'Caesar cipher'  

» DES  

» Triple-DES  

» AES 

# LIVE CODING

Funktionen um 'Caesar cipher' automatisch zu knacken.

# ASYMMETRISCHE KRYPTOSYSTEME (1/2)

Bestehend aus 2 Schlüsseln

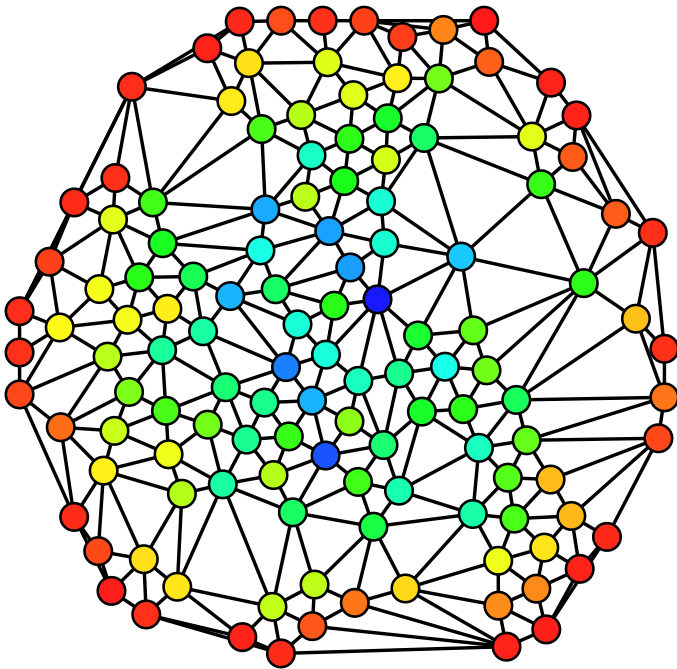
- › Public Key
  - › Öffentlich zugänglicher Schlüssel
  - › Ergänzt mathematische Operationen des Private Key.
  - › Verwendet zum **Verschlüsseln** von Nachrichten an Eigentümer des Private Key



# ASYMMETRISCHE KRYPTOSYSTEME (2/2)

- › Private Key
  - » Zum **Signieren** von Nachrichten
  - » Signatur: um Urheberschaft und Integrität zu prüfen
  - » Zum **Entschlüsseln** von verschlüsselten Nachrichten  
(zum zugehörigen Public Key)

# GRAPHENTHEORIE



Analysiert die Eigenschaften von Graphen und ihre Verbindungen zueinander.

- › Social Network Analysis
- › Verkehrsnetze
- › ...

# TECHNISCHE INFORMATIK

Beschäftigt sich mit der Hardware der Informatik zur Lösung verschiedenster Anforderungen wie

- › Echtzeitsysteme
- › Eingebettete Systeme
- › Mikroprozessoren
- › Rechnerarchitektur
- › Rechnerkommunikation

# ECHTZEITSYSTEME

Computer oder Systeme die in Echtzeit gewisse Werte überwachen/berechnen und bei Bedarf reagieren müssen.

- › Temperaturüberwachung
- › Eingabe in Computer-Terminals
- › Airbag-Steuerung und
- › ABS für Autos

# EINGEBETTETE SYSTEME

Sind Computer die in einen technischen Kontext eingebunden ist und im Hintergrund Arbeiten übernimmt.

- › Blu-Ray Player
- › Fernseher
- › Kühlschrank
- › Mobiltelefon
- › Board-Computer im Auto
- › ... Kleinstcomputer und ICs

# RECHNERARCHITEKTUR & MIKROPROZESSOREN

Design und Organisation von Rechnern

- › Ziel: Erstellung eines Bauplanes für einen Computer
  - » Architektur des Prozessors
  - » Design der Hauptplatine
  - » Verbund von Prozessor mit Arbeitsspeicher (BUS-System)
  - » Entwicklung von Speicherchips, Festplatten, Bildschirmen etc.

# RECHNERKOMMUNIKATION

Beschäftigt sich mit dem Datenaustausch zwischen verschiedenen Computern. Ein Rechnernetz stellt den Zusammenschluss mehrerer Computer (oder Sensoren/Agenten/Aktoren) dar.

- › Kommunikation über bekannte Protokolle
- › Aufgabe: Software und Hardware für effizienten Datenaustausch erstellen.
  - ›› Aufbau des Netzes
  - ›› Verwendete Protokolle

# PRAKTISCHE INFORMATIK

Beschäftigt sich mit konkreten und praktischen Problemen der Informatik wie

- › Programmiersprachen & Softwareentwicklungsprozess
- › Algorithmen
  - ›› Suchen und
  - ›› Sortieren von Daten
- › Datenstrukturen
- › Betriebssysteme
- › Datenbanken

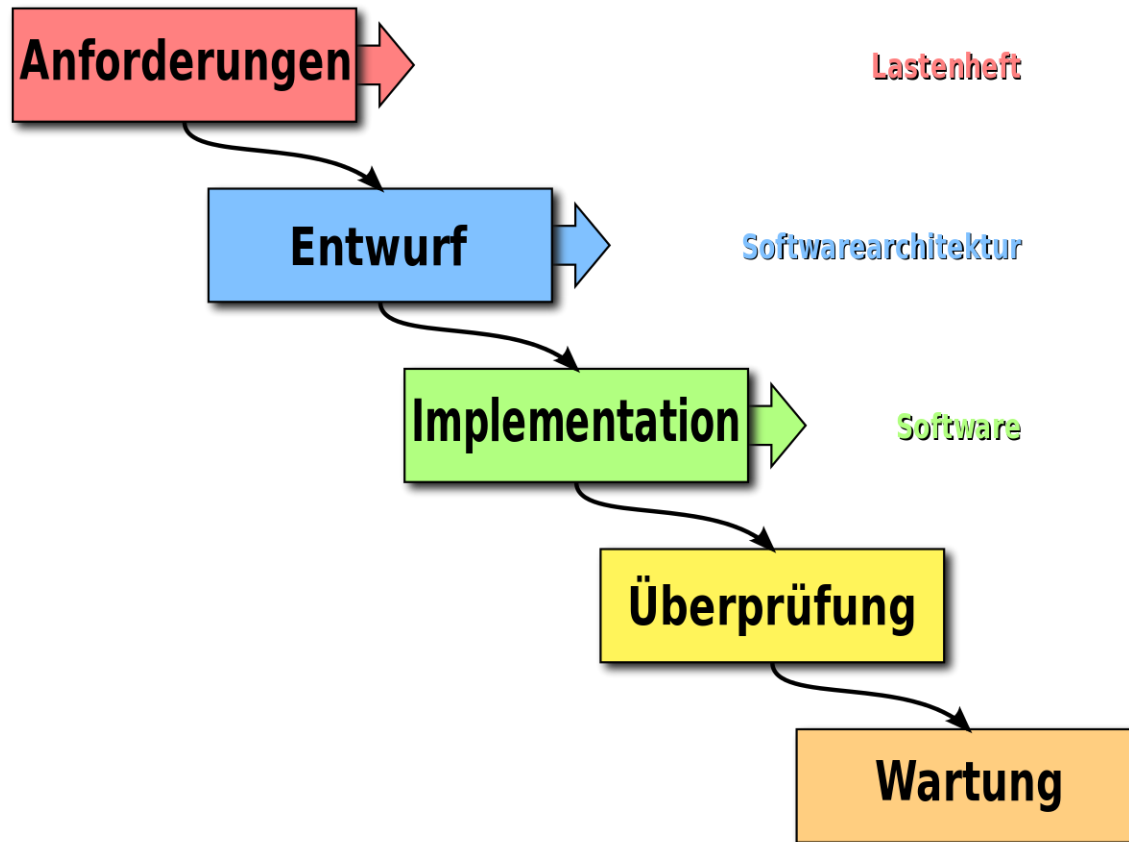


# PROGRAMMIERSPRACHEN

Entwicklung von Programmiersprachen die Menschen helfen dem Computer Anweisungen zu geben.

- › Compiler oder Interpreter übersetzen ein Programm in Maschinensprache.
- › Jede Programmiersprache hat eigene Compiler oder Interpreter.
- › Große Menge an Programmiersprachen mit vielen Unterschieden untereinander vorhanden (C, C++, Python, Ruby, Java etc.)

# SOFTWAREENTWICKLUNGSPROZESS



Von Paul Hoadley, Paul Smith and Shmuel Csaba Otto Traian, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=29119277>

# ALGORITHMEN

Ein Algorithmus beschreibt den Lösungsweg für ein Problem  
für (z.B.)

- › Sortieren von Daten
- › Suchen von Daten

Verschiedene Algorithmen arbeiten unterschiedlich effizient  
und benötigen unterschiedlich viele Ressourcen.

- › Viel Speicher, kurze Laufzeit vs.
- › Längere Laufzeit mit wenig Speicherbedarf

# DATENSTRUKTUREN

Legen fest, wie gewisse Daten gespeichert und darauf zugegriffen werden kann.

- › Beispiel: Stack (Stapelspeicher)
- › LIFO Prinzip
  - ›› Last In (Letztes drauf)
  - ›› First Out (Erstes weg)
- › Komplexe Datenstrukturen
  - ›› Bäume
  - ›› Graphen

# BETRIEBSSYSTEM

Ermöglicht das Verwenden des Computers

- › Verwaltet die Betriebsmittel (Hardware) wie Arbeitsspeicher, CPU, Ein-/Ausgabegeräte etc.
- › Management und Strategien für:
  - » Multiprocessing
  - » Arbeitsspeicher-Verwaltung
  - » Ein-/Ausgabegeräte Steuerung
  - » Prozess-Abläufe (Wer darf wann, was und wie lange machen!)

# DATENBANKEN

Elektronische Sammlung von Daten, die aus Benutzersicht zusammen gehört, strukturiert gespeichert.

- › Kontodatenbank einer Bank
- › Personaldatenbank einer Firma
- › Aufgaben: Schnell und zuverlässig auf große (zusammengehörige) Datensätze zugreifen.
- › z.B. auf alle Kunden die ein Konto nach dem 01.01.2016 bei einer bestimmten Filiale erstellt haben.

# ANGEWANDTE INFORMATIK

Die Angewandte Informatik beschäftigt sich mit der Erforschung und Entwicklung von Anwendungen von Rechnern wie

- › Grafische Datenverarbeitung
- › Datenbanksysteme
- › Numerik
- › Künstliche Intelligenz
- › Wirtschaftliche, kommerzielle Anwendungen
- › Technisch-wissenschaftliche Anwendungen

# WIRTSCHAFTLICHE, KOMMERZIELLE ANWENDUNGEN

Programme als Produkte

- › Buchhaltung
- › Rechnungswesen
- › Office-Suiten
- › Terminverwaltung
- › etc.



# TECHNISCH-WISSENSCHAFTLICHE ANWENDUNGEN

Software für die Durchführung von Simulationen

- › Ampelanlagen und Flugüberwachung

Anwendungen für numerische Probleme:

- › Effiziente Repräsentation von Daten

# INTERDISZIPLINÄRE INFORMATIK

Die Anwendung von erprobten Konzepten der Informatik auf  
Problemstellungen anderer Disziplinen

- › Grafische Datenverarbeitung
- › Biomedizinische Informatik
- › Computerlinguistik

# GRAFISCHE DATENVERARBEITUNG

Befasst sich mit der Erstellung, Bearbeitung und Erfassung von Bildern am Computer.

- › Computer Visualisierung
  - ›› Hilft dem Computer “zu sehen”
  - ›› Erkennen von Mustern in Videos etc.
- › Computer Grafik
  - ›› Erstellung und Bearbeitung von Bildern am Computer (= Computergrafiken)
  - ›› Schattierungen Berechnen, Animationen

# BIOMEDIZINISCHE INFORMATIK

Medizinische Problemstellungen mit  
Hilfe der Informatik zu lösen

- › Entschlüsselung von DNA
- › Früherkennung von Krankheiten
- › Ausbreitung von Krankheiten über Epidemie-Modelle (Ebola)
- › Vorhersage und Beratung der WHO über Informatiker

# COMPUTERLINGUISTIK

Untersucht ob und wie die natürliche Sprache mit dem Computer verarbeitet werden kann  
(Natural Language Processing).

- › Wichtige Worte aus Texten extrahieren (z.B. Orte, Namen, Datum)
- › Automatisches Übersetzen von Texten
- › Automatische Zusammenfassungen generieren
- › Kontexte erkennen und zusätzliche Informationen bereitstellen
- › Spracherkennung, Sprachsynthese

# DATENSTRUKTUREN

# DATENSTRUKTUREN

Dienen dem systematischen Ablegen und Aufrufen von Daten.

- › Speicherung
- › Organisation
- › Effizienz
- › regelt Art des Zugriffs

# DATENSTRUKTUREN

## BEISPIELE

- › Array
- › assoziatives Array (Dictionary)
- › Warteschlange (FiFo)
- › Stapelspeicher (LiFo)
- › Graphen
- › Bäume (Binärbaum)



# ARRAY

```
a = [1, "b", "III", 4, 5]
a[0]
a[2]
a[5]
```

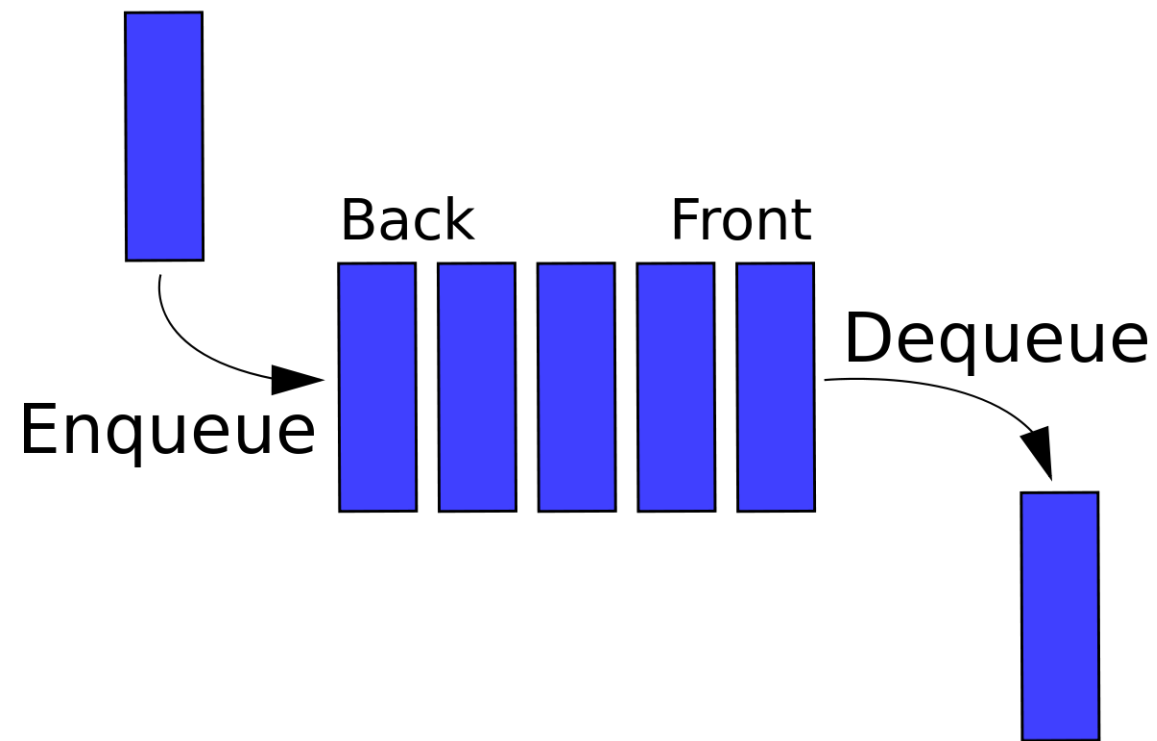
```
1
'III'
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
IndexError: list index out of range
```

# DICTIONARY

```
d = {"element1": 1, "myelement": "python", "python": 3.5}  
d['element1']  
d['python']  
d['myelement']
```

```
1  
3.5  
'python'
```

# WARTESCHLANGE (FIFO)



By This Image was created by User:Vegpuff. - Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=7586271>

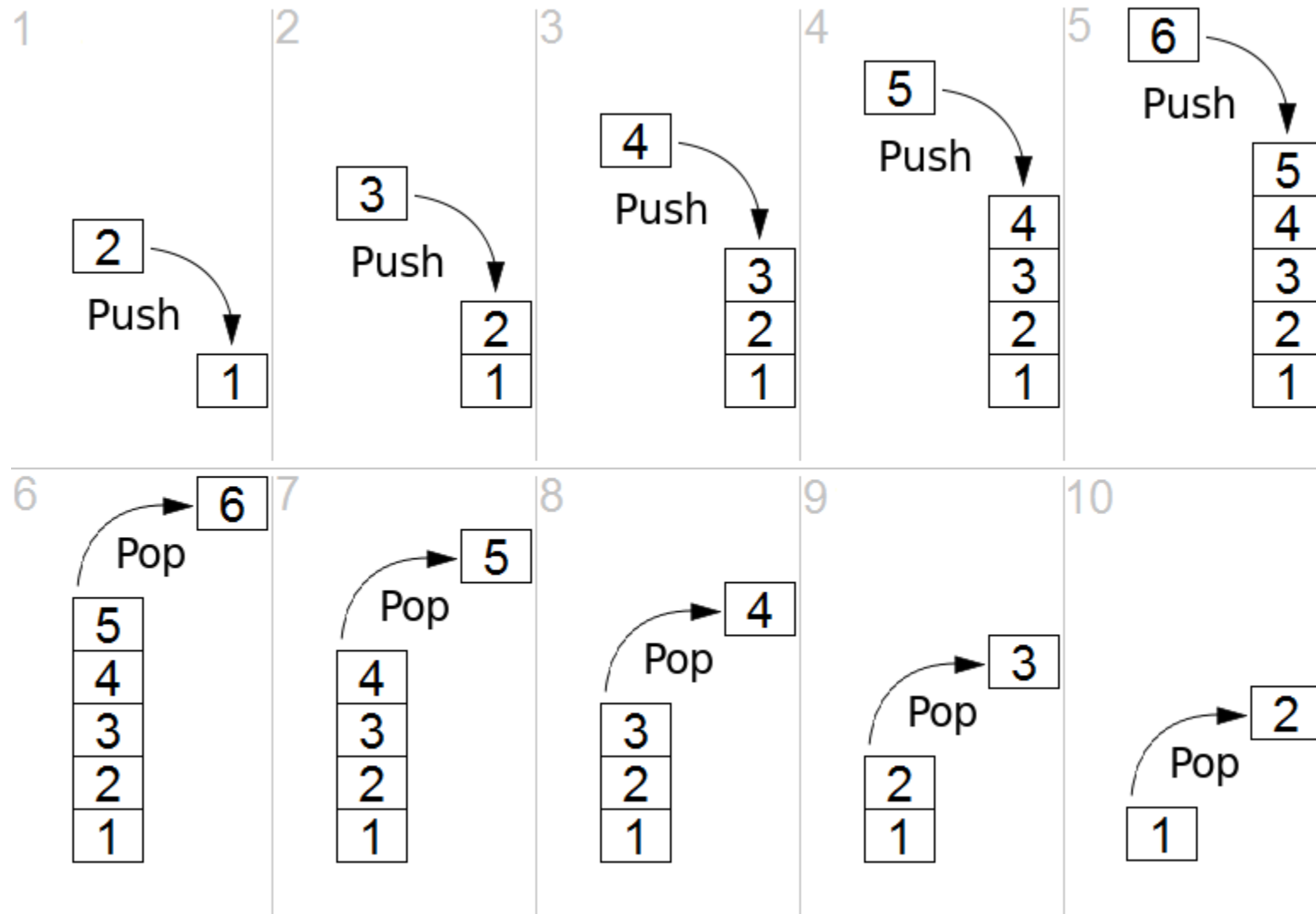
# WARTESCHLANGE (FIFO)

```
import queue
q = queue.Queue()
q.put(1)
q.put(2)
q.put("last")

q.get()
q.empty()
q.get()
q.get()
q.empty()
```

```
1
False
2
'last'
True
```

# STAPELSPEICHER (LIFO)



# STAPELSPEICHER (LIFO)

```
import queue
q = queue.LifoQueue()
q.put(1)
q.put(2)
q.put("last")

q.get()
q.empty()
q.get()
q.get()
q.empty()
```

```
'last'
False
2
1
True
```

# GRAPHEN

- › bestehen aus **Kanten** und **Knoten**
- › Eigenschaften:
  - » gerichtete Graphen: Kanten haben Richtung
  - » ungerichtete Graphen können in beide Richtungen 'begangen' werden.
  - » gewichtet: Kanten haben Gewicht
  - » zyklisch: Weg von Knoten A zurück zu A ohne eine Kante mehrfach zu gehen

# GRAPHENOPERATIONEN

- › Hinzufügen eines Knotens (mit oder ohne Kanten)
- › Entfernen des Knotens A, entfernt auch alle Kanten zu A
- › Es gibt keine Kanten ohne Knoten an beiden Enden



# BÄUME

Sonderform von Graphen

- › Bäume: zusammenhängende, azyklische Graphen
  - » gerichtet
  - » ungerichtet
  - » Binärbaum: maximal 2 Nachkommen pro Knoten

# FRAGEN?

# NÄCHSTES MAL

2016-11-16 16:00