

Informatik

Einführungsvorlesung

Prof. Dr.-Ing. Thomas Wiedemann
Fachgebiet Informatik / Mathematik



Überblick zur gesamten Veranstaltungsreihe

- Motivation - Warum sind EDV & Informatik für Sie relevant ?
- Einführung in Grundbegriffe der EDV & Informatik
 - Aufbau moderner Computer (Hardware)
 - Rechnen mit Bit und Bytes
 - Erstellung und Einsatz von Programmen (Software)
- **Algorithmen in Theorie und Praxis**
- **Programmierung mit Visual Basic**
- Entwurf moderner, grafischer Bedienoberflächen
- Parallel zur Programmierung :
 - Kriterien zur Auswahl von Hardware und Software
 - aktuelle Entwicklungen in der EDV
 - Probleme bei der numerischen Berechnung mit Computern
 - Kurzer Ausflug in andere Programmiersprachen (Java, HTML5, JavaScript)

Organisatorisches

- Folien sind im Internet verfügbar (Bitte auf den Seiten der Fak. Informatik / Mathematik unter “Personen – Wiedemann” nachschauen) oder direkt auf
<http://www.htw-dresden.de/~wiedem>
- (leider) nur 1 Semester ! (reicht nicht für größere Programme ...)
- überdurchschnittliche Ausführung von Praktikumsaufgaben wird mit virtuellen Pluspunkten vermerkt
- schriftliche Klausur am Ende des Semesters
- Wichtig sind Verständnis von Theorie und Praxis und kreative Anwendung des Gelernten ! (Demoklausur und Details am Ende des Semesters ...)

Info zur Symbolik

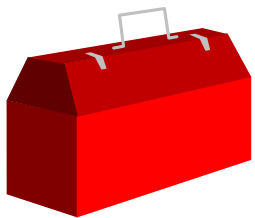


Ohne
Kennzeichnung

- Schwerpunktwissen (Definitionen/ absolute Grundlagen)
- katastrophale Auswirkungen bei Unkenntnis
- Standardwissen - **Verständnis** und ungefähre Wieder-
gabe mit eigenen Worten wird in Prüfung verlangt



- zusätzliche Information
- maximal Einfluß auf die Entscheidung zwischen „Gut“
und „Sehr gut“ bzw. „Ausgezeichnet“



- Werkzeugkasten : nicht direkt prüfungsrelevant (keine
direkten Fragen danach),
- sehr wichtig für die praktische Arbeit und damit
indirekter Einfluß auf den Zeitfaktor bei der Prüfung

Motivation - Warum sind EDV & Informatik für Sie relevant ?

Die Informatik (EDV) in Ihrem Beruf :

- über Rechenschieber, Taschenrechner zum Computer als allgemeines Arbeitsmittel
- Berechnung komplexer Aufgaben bei der Produktentwicklung
- Ablage und Verarbeitung von Daten in Datenbanken oder Dateien
- Planung und Optimierung von Produktionsprozessen
- Kommunikation (Email / Datenversand / Fernwartung)
- Computer als universelle Steuerungseinheit von Maschinen und Anlagen (immer mehr Programmierung statt traditioneller Verdrahtung)

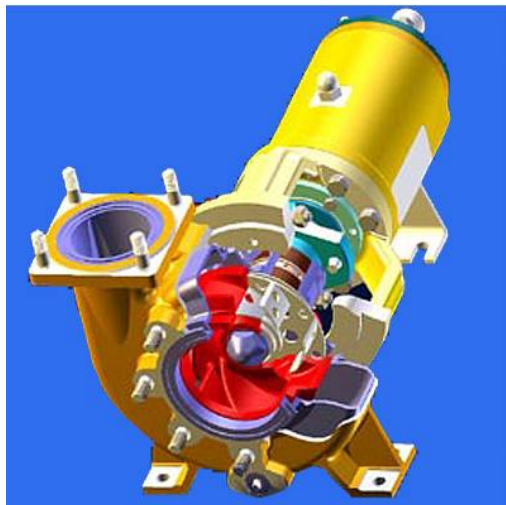


Die Beherrschung der Computertechnik wird ein notwendiges Handwerkszeug und entscheidet zwischen Profi und Durchschnitt !

Highlights der Computertechnik in Ihrem Beruf (I)

Parametrische CAD-Produktmodelle :

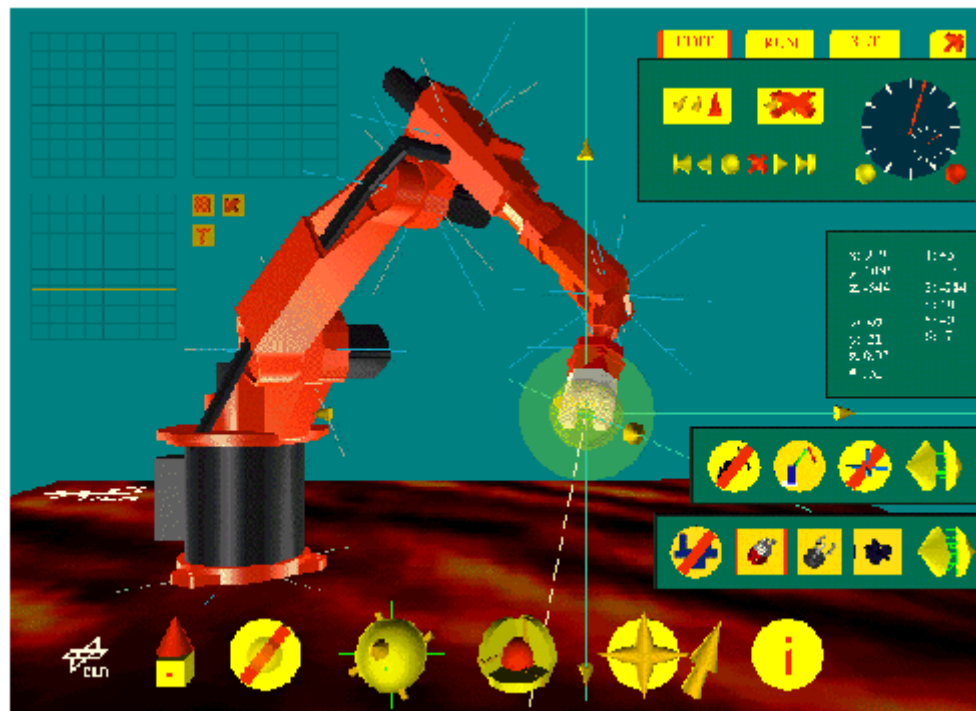
- Entwurf von kompletten Maschinen über **funktionale Abhängigkeiten**
- **Beispiel:** Entwurf eines Füllautomaten für 0,33l-Coladose mit funktionalen Abhängigkeiten aller Automatenparameter { Sirupdosierer = $f(\text{Dosenvolumen})$, Greifer = $f(\text{Dosenhöhe})$, Isolierung = $f(\text{Klimazone})$, Farbe = $f(\text{Kunde})$ }
- **Effekt:** 95% geringere Entwicklungszeiten bei ähnlichen Produkten !!!
- notwendig ist eine maschinell verarbeitbare Beschreibung der Abhängigkeiten durch kleine Formeln oder logische Bedingungen (Motivation für Programmierung)



Highlights der Computertechnik in Ihrem Beruf (II)

Simulation von Produkten und Anlagen :

- Crash-Simulation bei der Automobilentwicklung (Problem Elchtest ?!)
- genaue Analyse mechanischer, thermischer oder elektrischer Parameter von Teilen oder ganzen Anlagen mit 3D-Animation, zunehmend räumlich verteilte Darstellung
- Simulation neuer Prozessoren und Computer
- wesentliche schnellere Innovationszyklen - „Time to market“



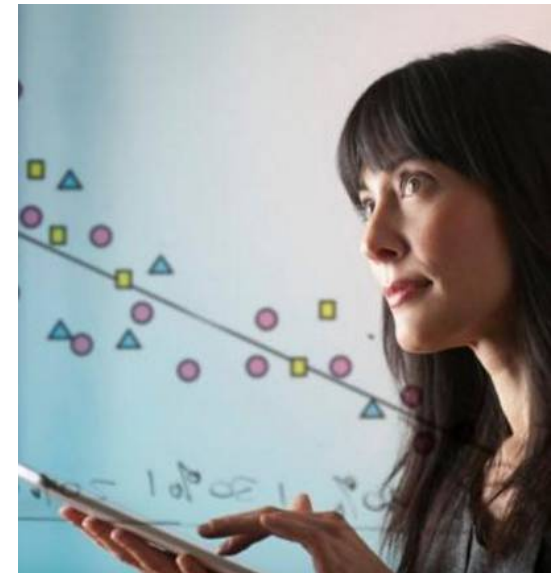
Highlights der Computertechnik in Ihrem Beruf (II)

Analyse von sehr großen Datenmengen – BIG DATA:

- Mit den Mitteln des Data Mining wird versucht Zusammenhänge zwischen den Daten zu finden und diese für zukünftige Geschäftsprozesse zu verwenden

Beispiele:

- Analyse von Kaufhauskassen (bzw. auch in Online-Shops):
 - zur Weihnachtszeit auffällige Häufung von gleichzeitigen Käufen (gleiche EC-Karte) von teuren Parfüm und Schmuck
 - => Last-Minute-Einkäufe von vergesslichen Männern am 23.12.
 - => Zusammenstellung von Kombiangeboten erhöht Umsatz
- Analyse von Fahrzeugbewegungsdaten (z.B. durch Google-Maps auf Smartphone) erlaubt Echtzeit Stauanzeigen
- Analyse von Facebook-Kommunikationen auf potentielle Shit-Storms oder Kunden-Produktvorlieben
- **Kritisch ist hier die Frage des Datenschutzes !**
- **=> Details in späteren VL**



Zusammenfassung zur Motivation

(Aus meiner persönlichen Sicht) **MÜSSEN SIE ALS WILING NICHT:**

- jede Codezeile einer Software schreiben oder verstehen
- in die letzten Bits der Hardware abtauchen

Sie sollten aber :

- das Zusammenspiel und Potentiale von Softwareanwendungen **bewerten** können
- neue Geschäftsfelder in Kombination mit neuen IT-Optionen erkennen können

Dazu müssen Sie

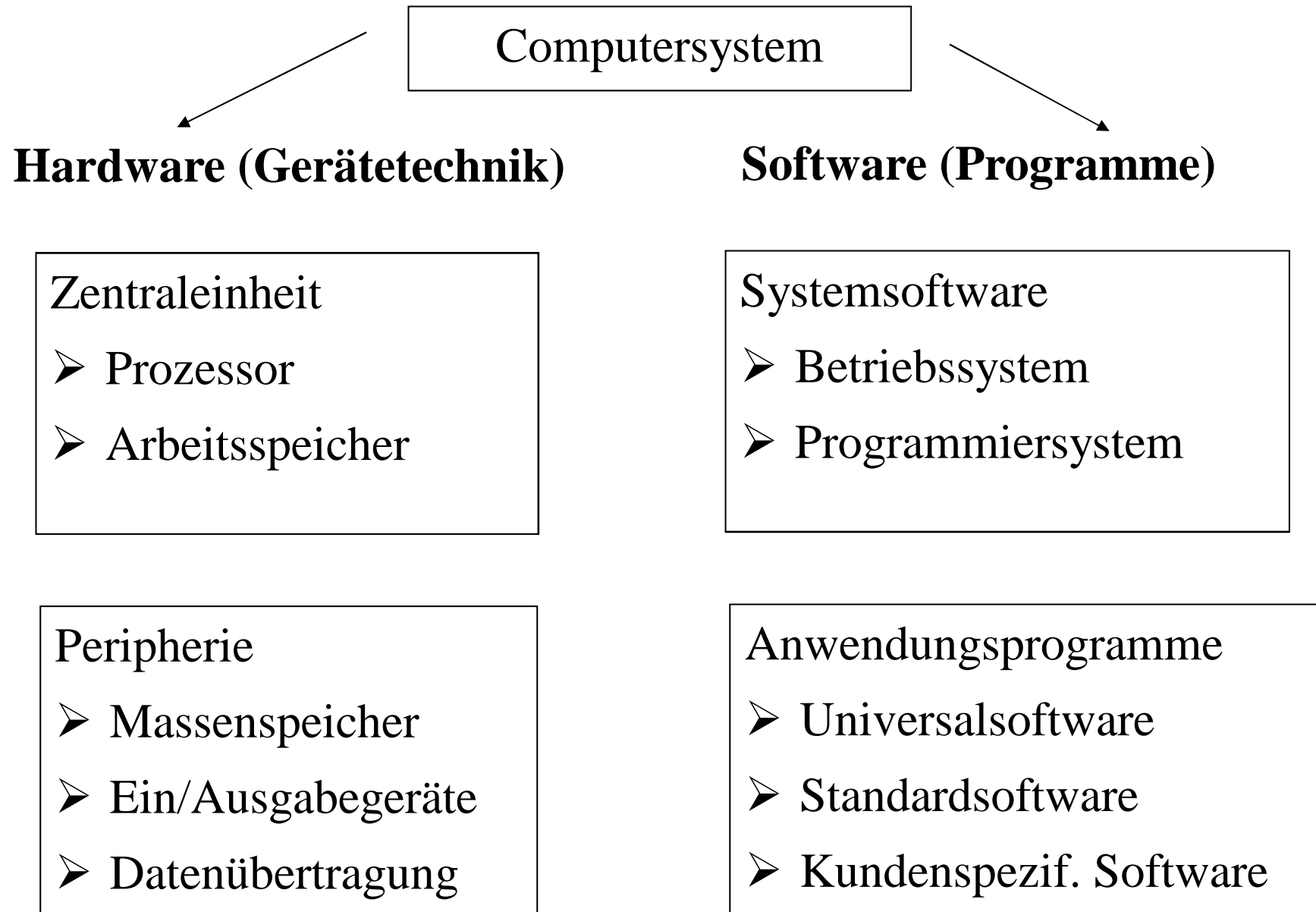
- grundlegende Zusammenhänge der Informatik kennen
- auch ein paar Prinzipien der Programmierung kennen, damit sie einfache Datenanalysen oder Schnittstellen zwischen vorhandenen Systemen erstellen können
- Die Informatik-Note ist bei vielen Bewerbungen ein relativ wichtiges Kriterium !





- **Informatik:** Theorie und praktische Methoden zur Verarbeitung von Informationen mit Computern
- **Elektronische Datenverarbeitung (EDV)** als ältere Bezeichnung mit einer etwas oberflächlicheren Bedeutung (... Rechnerbedienung)
- Die Informatik gliedert sich gegenwärtig in
 - **Theoretische Informatik** - Entwicklung und Analyse grundlegender Methoden und Vorschriften (Algorithmen) zur Informationsverarbeitung
 - **Technische Informatik** - Bereitstellung der Gerätebasis (physikalische Grundlagen, Schaltkreise, Computer, Rechnerarchitekturen, Vernetzung)
 - **Praktische Informatik** - Entwicklung von Programmen und Werkzeugen zur Problemlösung (Compiler, Grafiksysteme, Datenbanken)
 - **Angewandte Informatik:** Programme/Service für andere Wissensgebiete (CAD /CAM/CAX-Systeme, betriebliche Informationssysteme, Computergrafik, ...)
- Diese Lehrveranstaltung stellt einen Mix aus den Teilgebieten dar.

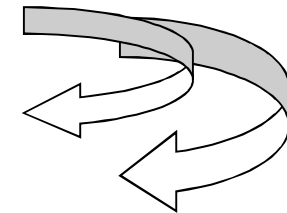
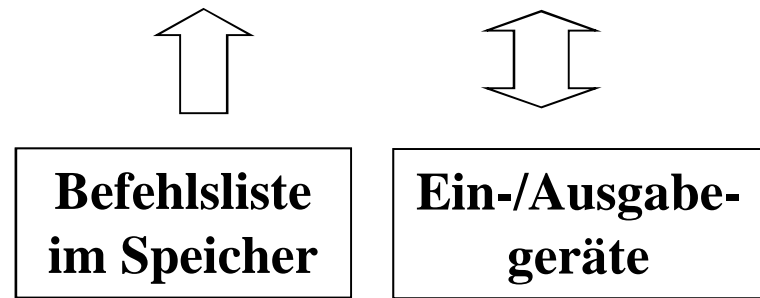
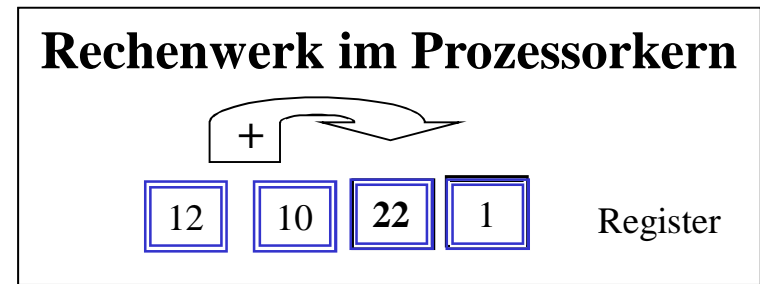
Aufbau vom Computersystemen



Arbeitsweise heutiger Computer

- **Prozessor:** Elektronik mit Fähigkeit zur schrittweisen Ausführung von sehr einfachen Operationen auf der Basis elektrischer Signale und logischer Schaltungen

- Hole Zahl A und B aus dem Arbeitsspeicher nach Register 1 und 2
- Addiere Zahl A und Zahl B
- Springe bei Ergebnis <0 zwei Schritte weiter, sonst einen !
- Schreibe 1 in Register 4, Springe 2 Schritte weiter
- Schreibe -1 in Register 4



- **Vorteile:** sehr schnell (bis 1 Milliarde Befehle / s)
- **Nachteile:** meist nur Befehle zur Operation mit Zahlen
- **Problem:** Wie erklären wir dem Prozessor komplexe Aufgaben ?

Der Algorithmusbegriff



Ein Algorithmus

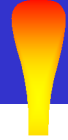
- beschreibt eine Methode, mit der eine Aufgabe gelöst wird,
- besteht in der Regel aus einer Folge von Schritten,
- führt bei korrekter Abarbeitung zur Lösung der Aufgabe,
- kann auch rein theoretisch oder umgangssprachlich beschrieben werden !

Die praktische Ausführung eines Algorithmus wird als **Prozeß** bezeichnet !

Algorithmen aus dem Alltag :

| Prozeß | Name des Algorithmus | Typische Schritte |
|-------------|----------------------|---|
| Stricken | Strickmuster | Zwei rechts, zwei links |
| Kochen | Rezept | Milch aufkochen, Kakao verrühren |
| Auto tunen | Tuning Guide | Zylinder aufbohren, Vergaser einstellen |
| Postversand | Tarifinformation | Eintüten, Messen, Wiegen, Stempeln |

Vom Algorithmus zum Programm



Aufgabe



Denken !!!!!!!!!!!!!!!

Algorithmus



Programmieren

**Maschinell lesbare Beschreibung
des Algorithmus
Quelltextprogramm**



Übersetzen (Compilieren)

**Algorithmus in maschinell
ausführbaren Anweisungen
Maschinenprogramm**



Ausführen (Starten)

Prozessor (Hardware)

Das Subjekt der jeweiligen Tätigkeit

SIE (Mensch / Informatiker)

SIE (Programmierer, Mensch ?)

Compiler (Übersetzungsprogramm)

**Mensch (manuell), Betriebssystem
(bei Automatikbetrieb)**

Die Entwicklung der Programmiersprachen

Aufgabe : Darstellung eines Algorithmus in maschinell lesbarer und ausführbarer Form, möglichst auch verständlich für den Menschen

Historie (zeitlich aufsteigend)

- direkte Eingabe eines **Kommandocodes**: *76,100* Addiere R1+Speicherzelle(100)
 - kaum für Menschen verständlich, extrem fehleranfällig, bis 1950 Standard !
 - nur seine **jeweils eigenen** Kommandocodes versteht der Prozessor !!!!!
- **Assemblercode**: *ADD R1,#Anzahl* (entspricht der 76 und 1e00 = Anzahl)
 - ab 1950, relativ verständlich, sehr aufwendig infolge expliziter Transportanweisungen
- **höhere Programmiersprachen** *Studenten+AnzahlGäste*
 - ab 1960, sehr effizient und verständlich, hoher Aufwand zur Übersetzung
- **Anwendungsorientierte Programmiersprachen und Bausteinsysteme**
 - ab 1985, stellen mächtige Befehle bereit : sendmail „Hi Tom“, „tom@web.de“
 - kommen teilweise ohne jegliche Programmierung aus, hohe Compilerkosten



Merke: Auch bei modernen Entwicklungssystemen und Programmiersprachen erfolgt zuletzt immer eine Erzeugung der primitiven Kommandocodes !

Verfügbare höhere Programmiersprachen

Entsprechend ihrer historischen Entwicklung zeitlich aufsteigend:

- FORTRAN : naturwissenschaftliche Berechnungen (sehr schnell)
- COBOL: ökonomische Berechnungen, auch heute tw. noch im Einsatz (bei Banken)
- BASIC: ehemals Anfängersprache, heute stark von Microsoft eingesetzt (VBA)
- PASCAL: sehr gut strukturierte Sprache
- C: stark auf Hardware und Betriebssystem bezogen
- C++ / C#: objektorientierte Erweiterung von C, **aktueller Industriestandard !**
- JAVA: rechnerunabhängige Sprache mit starkem Internetbezug

Erhältliche Compiler und Entwicklungssysteme:

- Visual Studio von Microsoft mit C++, C# JAVA und VisualBasic
- alle MS-Office Produkte sind ausgestattet mit Visual Basic for Applications (VBA)
- JBuilder von Borland für C++, Java, Borland-Delphi mit Object PASCAL
- diverse JAVA- und C++-Compiler von IBM, Symantec und anderen Firmen

Merke: Entgegen vieler (fanatischer) Diskussionen hat jede Sprache ihre Vor- und Nachteile und damit eine gewisse Daseinsberechtigung !!

Die Entwicklung von Algorithmen

Aufgabe



Denken !!!!!

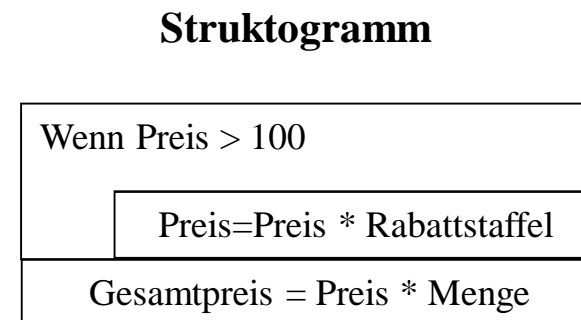
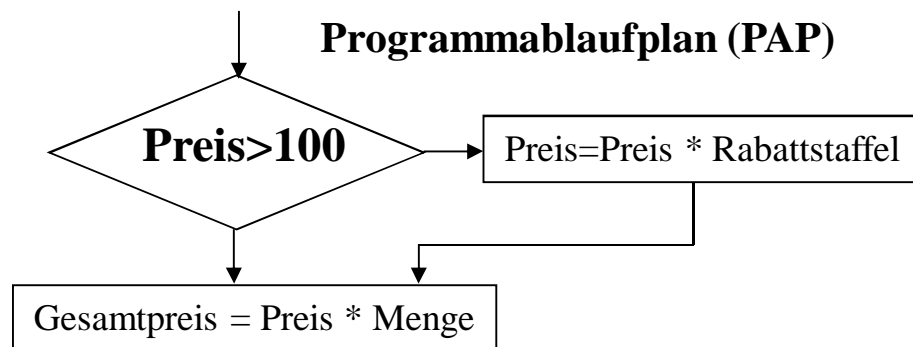
Algorithmus

1. Stufe : Gedankliche Erarbeitung

- entscheidet über die grundlegende Qualität des Programms
- kann alle nachfolgenden Stufen stark positiv oder negativ beeinflussen (ist teilweise eine Kunst ??? JA !!)

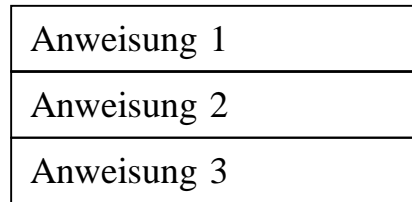
Gegenwärtig übliche Verfahren zur Unterstützung und Dokumentation :

- Beschreibung mit **normaler Sprache**: *Gebe Rabatt bei Preis größer 100 Euro*
 - sehr gut verständlich auch für Nicht-Informatiker (Kundenlevel, Ihr aktueller Level ?)
 - (noch) schlecht maschinell verarbeitbar, teilweise relativ lange Beschreibung
- **Pseudocode**: *WENN Preis > 100 DANN PREIS = PREIS * Rabattstaffel*
 - sehr gut lesbar, gut umsetzbar in Programme
- **grafische Darstellungen** (ausgezeichnet lesbar, noch schwierig umsetzbar):

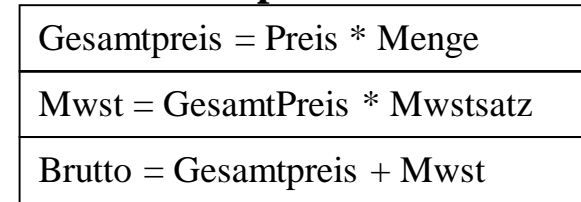


Grundlegende Strukturen von Algorithmen

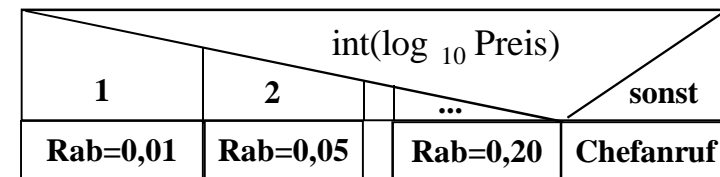
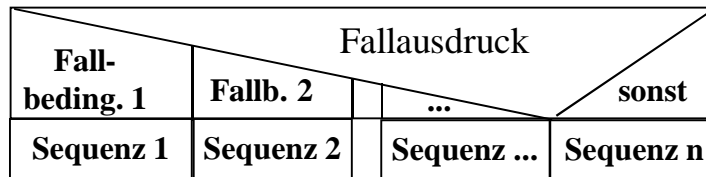
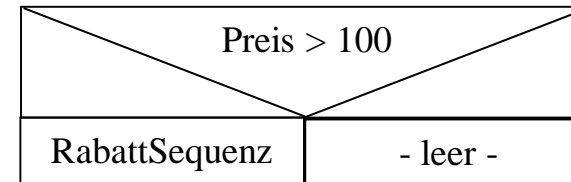
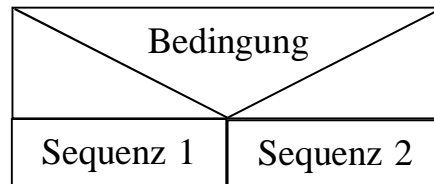
Sequenz
(Anweisungsfolge)



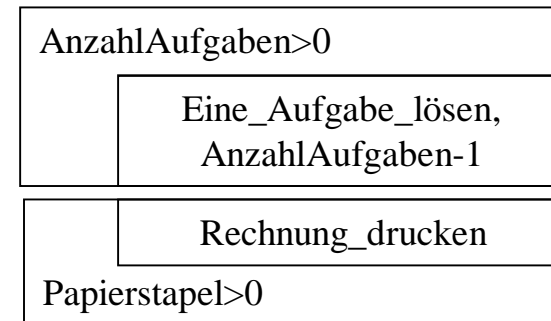
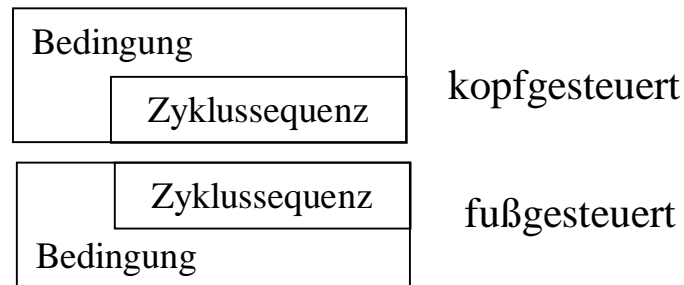
Beispiele



Selektion
(Auswahl von Alternativen)



Zyklus
(Iteration, Schleifen)



Merke: Alle lösbaren Aufgaben lassen sich auf diese Grundstrukturen unabhängig von Programmiersprachen abbilden .

Die Zukunft der Algorithmenbeschreibung ?

- **Intentional Programming** ist ein neues Projekt in der allg. Softwareentwicklung
- initiiert von Charles Simonyi (Entwicklungschef von Microsoft)
- orientiert auf Trennung der Anwendungsentwicklung von der Codeimplementierung
- Abhängigkeit zwischen Programmierung und Programmiersprache soll dadurch vollständig aufgehoben werden
- Datenbank zur Verwaltung von Abstraktionen
- Eingabe und Verwaltung dieser Abstraktionen über beliebige Editoren
- neben Quellcode auch grafische Notationen und auch Mischformen davon möglich

