

Informatik I

1. Grundlagen

Bernhard Nebel

Albert-Ludwigs-Universität Freiburg

22. Oktober 2013

Informatik I

22. Oktober 2013 — 1. Grundlagen

1.1 Inhalt der Vorlesung

1.2 Was ist Informatik?

1.3 Computer, Algorithmen, Programme,
Programmiersprachen und Prozesse

1.1 Inhalt der Vorlesung

Inhalt

Wir vermitteln in dieser Vorlesung Grundkenntnisse und Grundfähigkeiten in den Bereichen

- ▶ Programmierung (Python)
- ▶ Modellierung
- ▶ Entwicklung
- ▶ Analyse
- ▶ Hintergründe (Informatik-Geschichte, Berufsethik, ...)
- ▶ Grundlagen (Berechnungsmodelle, Programmiersprachenparadigmen, ...)
- ▶ Denken wie ein Informatiker/eine Informatikerin

1.2 Was ist Informatik?

Versuch der Definition I

Informatik-Duden

Wissenschaft von der systematischen Verarbeitung von Informationen, besonders der automatischen Verarbeitung mit Hilfe von Digitalrechnern (Computern).

Versuch der Definition II

Gesellschaft für Informatik

Das Wort **Informatik** setzt sich aus den Wörtern **Information** und **Automatik** zusammen und bezeichnet die Wissenschaft von der systematischen Verarbeitung von Informationen mit Hilfe von Rechenanlagen.

Aber:

Computer science is no more about computers than astronomy is about telescopes! (Dijkstra)

Versuch der Definition III

Association of Computing Machinery

Computer science and engineering is the systematic study of algorithmic processes—their theory, analysis, design, efficiency, implementation, and application—that describe and transform information. The fundamental question underlying all of computing is: What can be (efficiently) automated?

Einordnung

- ▶ Informatik hat etwas von Mathematik und ist insofern eine **Strukturwissenschaft**: Analyse von Strukturen

- ▶ Informatik hat etwas von Elektrotechnik und ist insofern eine **Ingenieurwissenschaft**: Design von Artefakten

Teilgebiete I (frei nach der GI)

Theoretische Informatik

Die Theoretische Informatik erforscht und entwickelt Konzepte zur Darstellung von Geräten und Prozessen als formal logische Systeme; damit ist sie die Grundlage für die Programmierung.

Die theoretische Informatik befasst sich insbesondere mit der Geschwindigkeit und dem Speicherverbrauch solcher Algorithmen.

Praktische Informatik

Die Praktische Informatik entwickelt grundlegende Lösungskonzepte für die wichtigsten Anwendungsbereiche der Informatik. Sie beschäftigt sich besonders mit der Entwicklung von Computerprogrammen mit Hilfe spezieller Programmiersprachen und deren Nutzung in großen Softwaresystemen.

Teilgebiete II (frei nach der GI)

Technische Informatik

Jedes Computersystem besteht aus drei funktional voneinander getrennten Einheiten: Dateneingabe, Datenbearbeitung und Datenausgabe. Die Entwicklung der hierfür erforderlichen Hardware ist der Kernbereich der Technischen Informatik.

Angewandte Informatik

Die Angewandte Informatik untersucht, inwieweit Abläufe durch den Einsatz von Computern automatisiert werden können. Verfahren der Simulation und Computergraphik, der Bild- und Sprachverarbeitung sowie der Modellierung schaffen konkrete Anwendungsmöglichkeiten für die Automatisierung.

Teilgebiete III (frei nach der GI)

Informatik und Gesellschaft

Der Bereich Informatik und Gesellschaft umfasst Soziologie, Philosophie, Jura und Politologie und ermöglicht eine umfassende Technikfolgenabschätzung für Computeranwendungen in der modernen Gesellschaft. Themen sind etwa Datenschutz, Softwarepatente, gesellschaftliche Bewegungen wie Open Source und ihr Verhältnis zum Urheberrecht.

Exkurs: GI ...



GI

Die **Gesellschaft für Informatik e.V. (GI)** ist die größte Vereinigung von Informatikerinnen und Informatikern im deutschsprachigen Raum. Sie versteht sich als Plattform für Informatikfachleute aus Wissenschaft und Wirtschaft, Lehre und Öffentlicher Verwaltung und versammelt eine geballte Konzentration an Wissen, Innovation und Visionen. Rund 20.000 persönliche Mitglieder, darunter 1.500 Studierende und knapp 300 Unternehmen und Institutionen, profitieren von unserem Netzwerk.

Exkurs: ACM ...



ACM

ACM (Association for Computing Machinery), the world's largest educational and scientific computing society, delivers resources that advance computing as a science and a profession. ACM provides the computing field's premier Digital Library and serves its members and the computing profession with leading-edge publications, conferences, and career resources.

1.3 Computer, Algorithmen, Programme, Programmiersprachen und Prozesse

- Computer
- Algorithmen und Kochen
- Programme und Programmiersprachen
- Berechnungsprozess
- Schluss

Computer ...

- ▶ Wie tauch(t)en Computer in unserem täglichen Leben auf?



- ▶ Kann man den Begriff präzise definieren?

Was ist ein Computer?

- ▶ **Informatik Duden:** „(engl.: to compute = rechnen, berechnen; ursprünglich aus dem lat. computare = berechnen ...): Universell einsetzbares Gerät zur automatischen Verarbeitung von Daten.“
- ▶ Die prinzipiellen Fähigkeiten und Beschränkungen von idealisierten Computern werden durch das Automatenmodell der **universellen Turing-Maschine** beschrieben (→ Theoretische Informatik).
- ▶ Der prinzipielle technische Aufbau eines heutigen Computers wird gut durch die **von-Neumann-Architektur** beschrieben (→ Technische Informatik).

Was tut ein Computer?

Um uns dieser Frage zu nähern, sollten wir vier Konzepte verstehen und unterscheiden:

- ▶ **Input/Output**,
- ▶ **Algorithmus**,
- ▶ **Programm**,
- ▶ **(Berechnungs)prozess**.

Eine hilfreiche Analogie ist das Kochen . . .

Input/Output

Eingabe (Input):



Ausgabe (Output):



Hier interessiert nur:

- ▶ Welche Zutaten stehen zur Verfügung?
- ▶ Wie schmeckt die fertige Pizza?

Algorithmus

- ▶ Wie wird die Pizza zubereitet?
- ▶ Ich folge einem Rezept (= **Algorithmus**).
- ▶ Wenn ich die Reihenfolge, in der die Paprika und die Pilze auf den Teig gelegt werden, ändere, ist das ein anderer Algorithmus, auch wenn das den Geschmack der Pizza vielleicht nicht beeinflusst.

Unsere Vorstellung vom Kochen

Die Analogie hinkt vielleicht ein wenig:

- ▶ Kochrezepte sind meistens nicht „idiotensicher“ genug. Sie lassen Freiheiten, und sie setzen manches Wissen voraus.
- ▶ Die meisten Rezepte sind für festgelegte Mengen von festgelegten Zutaten geschrieben.

Parametrisierte Rezepte sind näher an Algorithmen ...

Parametrisierte Rezepte: Art der Zutaten



Obstkuchen

Zutaten

4 mittelgroße Eier, 1 Tasse Zucker, [...] **Obst nach Saison, z.B. Beeren oder Mandarinenfilets**

Zubereitung

[...] Den Pudding auf dem Kuchen verteilen und mit **Obst nach Saison** belegen. [...]

Parametrisierte Rezepte: Menge der Zutaten

1. Sie braten Pfannkuchen: die Zubereitung des Teiges ist unabhängig von der Menge der Zutaten exakt dieselbe.
2. Dann folgt etwas wie: „Solange die Teigschüssel nicht leer ist, tauche die Kelle in die Schüssel, schütte den Inhalt in die heiße Pfanne, ...“
3. Wenn Sie sehr große Zutatenmengen haben, stimmt auch ❶ nicht mehr: Sie werden eine größere Schüssel brauchen oder den Teig in mehreren Fuhren machen müssen.
4. Das klingt doch schon nach einem ganz spannenden **Algorithmus!**

Algorithmus

Vorschrift zur Durchführung einer Berechnung (Folge von Einzelschritten) mit folgenden Eigenschaften:

- Effektivität** Jeder Einzelschritt ist ausführbar.
- Determiniertheit** Der nächste Einzelschritt ist stets festgelegt.
- Fintheit** Die Vorschrift ist endlich.
- Terminierung** Die Berechnung endet nach endlich vielen Einzelschritten – für alle legalen Eingaben.
- Generalität** Die Vorschrift kann eine ganze Klasse von Problemen lösen.
- Präzision** Die Bedeutung jedes Einzelschritts ist eindeutig festgelegt.

Vorschrift, Einzelschritt

- ▶ Es geht hier um den **Inhalt** der Vorschrift, wie auch immer sie notiert ist.
- ▶ Wir verstehen noch nicht genau, was ein „Einzelschritt“ sein könnte. Das macht nichts!
- ▶ Erst recht wissen wir nicht, wie ein Einzelschritt notiert werden könnte. Auch das macht nichts!
- ▶ Entscheidend ist hier nur: **Irgendwie** wird ein Einzelschritt notiert, und wenn der Computer (oder unser Koch) dies liest, weiß er genau, was er zu tun hat.

Weitere Beispiele für Algorithmen

(aus Mathematik und Alltag)

- ▶ Schriftliche Addition, Multiplikation, Division
- ▶ Lösen von linearen Gleichungen
- ▶ Wegbeschreibungen
- ▶ Bedienungsanleitungen, Spielregeln

Und nochmals: **Irgendwie** muss man Algorithmen notieren, wenn man sie einem Computer oder Menschen kommunizieren will, aber die Notation ist nicht der Algorithmus.

Programm

Ein **Programm** ist der Algorithmus notiert („aufgeschrieben“) in einer geeigneten Sprache.



Es gibt verschiedene Programmiersprachen, aber sie alle sind **formale** Sprachen, d.h., sie sind **exakt**, durch strikte Regeln, definiert. Das unterscheidet sie von natürlichen Sprachen wie Deutsch oder Italienisch.

Programmiersprachen

- ▶ Systemprogrammiersprachen
 - ▶ Nah an der Maschine
 - ▶ Abbildung auf Maschine offensichtlich
- ▶ Höhere Programmiersprachen
 - ▶ Idealisieretes Berechnungsmodell
 - ▶ Abbildung auf Maschine einfach
- ▶ Deklarative Programmiersprachen
 - ▶ Einfacheres Berechnungsmodell
 - ▶ Abbildung auf Maschine schwierig

Elemente von Programmiersprachen

So wie **Sätze** in natürlicher Sprache aus **Wörtern** und **Satzzeichen** gemäß einer bestimmten **Grammatik** zusammengefügt werden, so werden **Programme** in einer Programmiersprache aus **Grundbausteinen** unter Verwendung von **Kombinationsmitteln** zusammengefügt.
Es kommt noch ein Konzept hinzu: **Abstraktionsmittel**, um Programmstücke zu benennen.

Prozess



Der Vorgang des Kochens, also das Ausführen eines Programms, an einem bestimmten Ort zu einer bestimmten Zeit.

Berechnungsprozess

- ▶ Der Ablauf eines Programms auf einem bestimmten Rechner zu einer bestimmten Zeit.
- ▶ In dieser Vorlesung spielt der Begriff des Prozesses keine große Rolle, obwohl wir natürlich unsere Programme auch gelegentlich mal laufen lassen wollen.
- ▶ In **Betriebssystemen** dreht sich alles um Prozesse. Z. B.: Wieviel Rechenzeit auf welchem Prozessor bekommt welcher Prozess wann spendiert?

Input/Output, Algorithmus, Programm, (Berechnungs)prozess

- ▶ Ein Algorithmus ist eine Vorschrift zur Durchführung einer Berechnung.
- ▶ Ein bestimmtes Input/Output-Verhalten kann evtl. durch verschiedene Algorithmen erreicht werden.
- ▶ Ein Programm ist die konkrete Umsetzung eines Algorithmus in einer Programmiersprache.
- ▶ Ein Algorithmus kann in verschiedenen Programmiersprachen und durch verschiedene Programme implementiert werden.
- ▶ Ein Programm kann mehrmals auf verschiedenen Computern auf der ganzen Welt laufen, gehört also zu vielen Prozessen.