

Informatik I: Einführung in die Programmierung

1. Grundlagen

Albert-Ludwigs-Universität Freiburg



UNI
FREIBURG

Prof. Dr. Peter Thiemann

16. October 2024

Inhalt der Vorlesung

Inhalt



UNI
FREIBURG

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)
- Modellierung

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)
- Modellierung
- Programmentwicklung

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)
- Modellierung
- Programmentwicklung
- Problemanalyse

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)
- Modellierung
- Programmentwicklung
- Problemanalyse
- Grundlagen (Berechnungsmodelle, Programmiersprachenparadigmen, ...)

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Die Vorlesung vermittelt Grundkenntnisse und Grundfähigkeiten in den Bereichen

- Programmierung (Python)
- Modellierung
- Programmentwicklung
- Problemanalyse
- Grundlagen (Berechnungsmodelle, Programmiersprachenparadigmen, ...)
- Denken wie ein Informatiker/eine Informatikerin

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Was ist Informatik?

Versuch der Definition I



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Informatik-Duden

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

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.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



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)

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



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



UNI
FREIBURG

- Informatik beschäftigt sich mit der Analyse von Strukturen und ist insofern eine **Strukturwissenschaft**

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Einordnung



UNI
FREIBURG

- Informatik beschäftigt sich mit der Analyse von Strukturen und ist insofern eine **Strukturwissenschaft**
- verwandt mit der Mathematik; verwendet die Sprache der Mathematik

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Einordnung



- Informatik beschäftigt sich mit der Analyse von Strukturen und ist insofern eine **Strukturwissenschaft**
 - verwandt mit der Mathematik; verwendet die Sprache der Mathematik
-
- Informatik beschäftigt sich mit der Planung und dem Entwurf von Artefakten und ist insofern eine **Ingenieurwissenschaft**

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Exkurs:
Think like a computer scientist

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.

- Was ist **berechenbar**?
- P = NP?

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Teilgebiet II (frei nach der GI)



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.

```
package org.eclipse.van.mercurius.begleiter;

public class BegleiterController extends BegleiterController implements
    BegleiterControllerListener, BegleiterController {
    private BegleiterControllerListener listener;
}

public void setControllerListener(BegleiterControllerListener listener) {
    this.listener = listener;
}

public void update() {
    if (listener != null) {
        listener.update();
    }
}

public void updateFromBegleiter(Begleiter begleiter) {
    if (listener != null) {
        listener.updateFromBegleiter(begleiter);
    }
}
```

Inhalt der Vorlesung

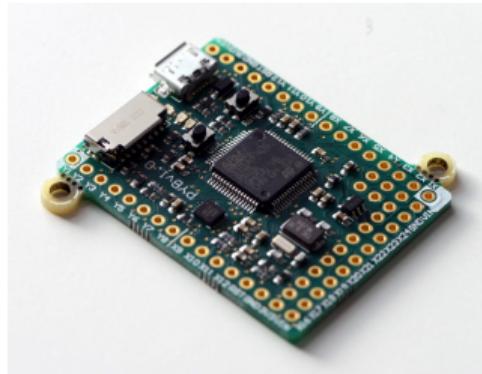
Was ist Informatik?

Algorithmus

Exkurs:
Think like a computer scientist

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.



Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Angewandte Informatik

Die Angewandte Informatik untersucht, inwieweit Abläufe durch den Einsatz von Computern automatisiert werden können. Hierunter fallen Verfahren der Simulation und Computergraphik, der Bild- und Sprachverarbeitung, maschinelles Lernen, KI, Robotik.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Exkurs:
Think like a computer scientist



Informatik und Gesellschaft

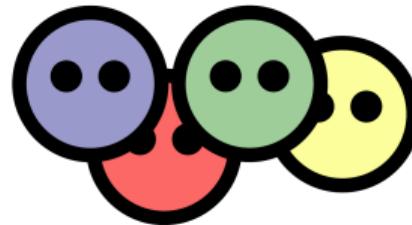
Der Bereich Informatik und Gesellschaft umfasst Soziologie, Philosophie, Jura und Politologie und ermöglicht eine umfassende Technikfolgenabschätzung für Computeranwendungen in der modernen Gesellschaft. Themen sind etwa Datenschutz, Softwarepatente, gesellschaftliche Bewegungen wie Open Source und ihr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

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist





Exkurs: GI



UNI
FREIBURG

GI

Die **Gesellschaft für Informatik e.V. (GI)** ist die größte und wichtigste Fachgesellschaft für Informatik im deutschsprachigen Raum. Sie vertritt die Interessen der Informatikerinnen und Informatiker in Wissenschaft, Wirtschaft, öffentlicher Verwaltung, Gesellschaft und Politik. Werden Sie Teil dieses lebendigen Netzwerks!

Die GI vernetzt rund 20.000 persönliche Mitglieder, darunter 1.500 Studierende und rund 250 Unternehmen und Institutionen miteinander.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Exkurs:
Think like a computer scientist

Mitgliedschaft

Kostenlos für Studenten!



ACM

Association for Computing Machinery (ACM), 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.

Membership

USD 19 / year for students. Mandatory if you want to become a CS researcher.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Computer, Algorithmen, Programme, Programmiersprachen und Prozesse

Etymology

According to the [Oxford English Dictionary](#), the first known use of *computer* was in a 1613 book called *The Yong Mans Gleanings* by the English writer [Richard Braithwait](#): "I haue [sic] read the truest computer of Times, and the best Arithmetician that euer [sic] breathed, and he reduceth thy dayes into a short number." This usage of the term referred to a [human computer](#), a person who carried out calculations or computations. The word continued with the same meaning until the middle of the 20th century. During the latter part of this period women were often hired as computers because they could be paid less than their male counterparts.^[1] By 1943, most human computers were women.^[2]

The [Online Etymology Dictionary](#) gives the first attested use of *computer* in the 1640s, meaning 'one who calculates'; this is an "agent noun from compute (v.)". The [Online Etymology Dictionary](#) states that the use of the term to mean "'calculating machine' (of any type) is from 1897." The [Online Etymology Dictionary](#) indicates that the "modern use" of the term, to mean 'programmable digital electronic computer' dates from "1945 under this name; [in a] theoretical [sense] from 1937, as *Turing machine*".^[3]



A human computer, with microscope and calculator, 1952

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

■ Filmtipp: Hidden Figures – Unerkannte Heldeninnen

Computer ...



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

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



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 **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:

- Ein-/Ausgabe,
- Algorithmus,
- Programm,
- (Berechnungs)prozess.

Eine hilfreiche Analogie ist das Kochen ...

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus
Computer

Algorithmen und
Kochen
Beispiel
Eigenschaften
Programme und
Programmierspra-
chen
Berechnungspro-
zess
Schluss

Exkurs:
Think like a
computer
scientist

Ein-/Ausgabe



UNI
FREIBURG

Eingabe:



Ausgabe:



Hier interessiert nur:

- Welche Zutaten stehen zur Verfügung?
- Wie sieht die fertige Pizza aus?

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Algorithmus



UNI
FREIBURG



Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

- Wie wird die Pizza zubereitet?

Algorithmus



UNI
FREIBURG

- Wie wird die Pizza zubereitet?
- Ich folge einem Rezept (\approx **Algorithmus**).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Algorithmus



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

- Wie wird die Pizza zubereitet?
- Ich folge einem Rezept (\approx 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 ist nicht perfekt:

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

Tatsächlich ist das Konzept eines Algorithmus nicht für die Zubereitung von Pizzen sondern für die **Durchführung einer Berechnung** entwickelt worden (geht zurück auf **Muhammed al-Chwarizmi** (ca. 780-850)).

In Deutschland massgeblich im Rechenbuch von **Adam Riese** (ca. 1492-1559).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer
Algorithmen und
Kochen

Beispiel

Eigenschaften
Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Algorithmus

- 1 Setze P auf 0.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Algorithmus

- 1 Setze P auf 0.
- 2 Falls $R = 0$, gebe P als Ergebnis zurück.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Algorithmus

- 1 Setze P auf 0.
- 2 Falls $R = 0$, gebe P als Ergebnis zurück.
- 3 Addiere L zu P hinzu.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Algorithmus

- 1 Setze P auf 0.
- 2 Falls $R = 0$, gebe P als Ergebnis zurück.
- 3 Addiere L zu P hinzu.
- 4 Reduziere R um 1.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Multiplikation zweier natürlicher Zahlen mit Hilfe der Addition und Subtraktion



Eingabe und Ausgabe

Eingabe: Zwei natürliche Zahlen L und R

Ausgabe: Das Produkt von L und R

Algorithmus

- 1 Setze P auf 0.
- 2 Falls $R = 0$, gebe P als Ergebnis zurück.
- 3 Addiere L zu P hinzu.
- 4 Reduziere R um 1.
- 5 Mache bei Schritt 2 weiter.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

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

Präzision

Die Bedeutung jedes Schritts ist eindeutig festgelegt.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

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

Präzision
Effektivität

Die Bedeutung jedes Schritts ist eindeutig festgelegt.
Jeder Schritt ist ausführbar.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

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

Präzision

Die Bedeutung jedes Schritts ist eindeutig festgelegt.

Effektivität

Jeder Schritt ist ausführbar.

Finitheit (statisch)

Die Vorschrift ist ein endlicher Text.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

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

Präzision

Die Bedeutung jedes Schritts ist eindeutig festgelegt.

Effektivität

Jeder Schritt ist ausführbar.

Finitheit (statisch)

Die Vorschrift ist ein endlicher Text.

Finitheit (dynamisch)

Zur Ausführung wird nur endlich viel Speicher benötigt.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

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

Präzision

Die Bedeutung jedes Schritts ist eindeutig festgelegt.

Effektivität

Jeder Schritt ist ausführbar.

Finitheit (statisch)

Die Vorschrift ist ein endlicher Text.

Finitheit (dynamisch)

Zur Ausführung wird nur endlich viel Speicher benötigt.

Terminierung

Die Berechnung endet nach endlich vielen Schritten – für alle legalen Eingaben.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Gegenbeispiele



- Male ein Haus (Präzision).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Gegenbeispiele



UNI
FREIBURG

- Male ein Haus (Präzision).
- Teile die Zahl durch 0, nenne die letzte Stelle der Zahl π (Effektivität).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Gegenbeispiele



UNI
FREIBURG

- Male ein Haus (Präzision).
- Teile die Zahl durch 0, nenne die letzte Stelle der Zahl π (Effektivität).
- Unendlich lange Vorschriften sind schwer vorstellbar, aber in der Mathematik gibt es unendliche Axiomensysteme (statische Finitheit).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer
Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Gegenbeispiele



UNI
FREIBURG

- Male ein Haus (Präzision).
- Teile die Zahl durch 0, nenne die letzte Stelle der Zahl π (Effektivität).
- Unendlich lange Vorschriften sind schwer vorstellbar, aber in der Mathematik gibt es unendliche Axiomensysteme (statische Finitheit).
- Gib alle Nachkommastellen der Zahl π an (dynamische Finitheit, Effektivität).

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer
Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Gegenbeispiele



- Male ein Haus (Präzision).
- Teile die Zahl durch 0, nenne die letzte Stelle der Zahl π (Effektivität).
- Unendlich lange Vorschriften sind schwer vorstellbar, aber in der Mathematik gibt es unendliche Axiomensysteme (statische Finitheit).
- Gib alle Nachkommastellen der Zahl π an (dynamische Finitheit, Effektivität).
- Ersetze den Test $R = 0$ durch $L = 0$ (Terminierung nur noch wenn $L = 0!$).

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Weitere wünschenswerte Eigenschaften



Weitere Eigenschaften, die oft als wünschenswert für einen Algorithmus angesehen werden:

Determinismus

Die Folgeschritte sind immer eindeutig festgelegt.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Weitere wünschenswerte Eigenschaften



Weitere Eigenschaften, die oft als wünschenswert für einen Algorithmus angesehen werden:

Determinismus

Die Folgeschritte sind immer eindeutig festgelegt.

Determiniertheit

Bei gleicher Eingabe erzeugt die Vorschrift die gleiche Ausgabe – berechnet also eine Funktion.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Weitere wünschenswerte Eigenschaften



Weitere Eigenschaften, die oft als wünschenswert für einen Algorithmus angesehen werden:

Determinismus

Die Folgeschritte sind immer eindeutig festgelegt.

Determiniertheit

Bei gleicher Eingabe erzeugt die Vorschrift die gleiche Ausgabe – berechnet also eine Funktion.

Generalität

Die Vorschrift kann eine Klasse von Problemen lösen.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer
Algorithmen und Kochen
Beispiel

Eigenschaften

Programme und Programmiersprachen
Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Weitere wünschenswerte Eigenschaften



Weitere Eigenschaften, die oft als wünschenswert für einen Algorithmus angesehen werden:

Determinismus Die Folgeschritte sind immer eindeutig festgelegt.

Determiniertheit Bei gleicher Eingabe erzeugt die Vorschrift die gleiche Ausgabe – berechnet also eine Funktion.

Generalität Die Vorschrift kann eine Klasse von Problemen lösen.

Alle Beispiele, die wir in dieser Vorlesung kennen lernen werden, erfüllen die Bedingungen. Aber auch Vorschriften, die diese Extra-Bedingungen nicht erfüllen, werden als Algorithmen angesehen.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer
Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

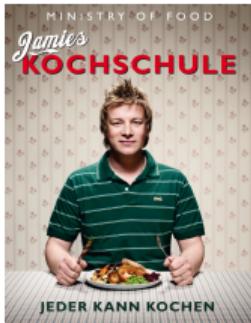
Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Programm

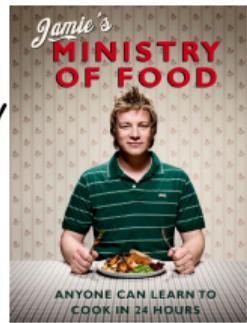
Ein **Programm** ist ein Algorithmus aufgeschrieben in einer geeigneten Sprache.



\neq



\neq



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.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Programmiersprachen



- Systemprogrammiersprachen
 - Nah an der Maschine
 - Abbildung auf Maschine offensichtlich

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

■ Systemprogrammiersprachen

- Nah an der Maschine
- Abbildung auf Maschine offensichtlich

■ Höhere Programmiersprachen

- Idealisiertes Berechnungsmodell (notional machine)
- Abbildung auf Maschine einfach

- Systemprogrammiersprachen
 - Nah an der Maschine
 - Abbildung auf Maschine offensichtlich
- Höhere Programmiersprachen
 - Idealisiertes Berechnungsmodell (notional machine)
 - Abbildung auf Maschine einfach
- Deklarative Programmiersprachen
 - Spezifikation der Aufgabe anstelle eines Berechnungsmodells (Was statt Wie)
 - Abbildung auf Maschine anspruchsvoll



Syntax

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.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist



Syntax

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.

Dazu kommen **Abstraktionsmittel**, um Programmelemente zu benennen.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Syntax

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.

Dazu kommen **Abstraktionsmittel**, um Programmelemente zu benennen.

Semantik

Bedeutung eines Satzes bzw. einen Programmstücks.

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist



Syntax

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.

Dazu kommen **Abstraktionsmittel**, um Programmelemente zu benennen.

Semantik

Bedeutung eines Satzes bzw. einen Programmstücks.

$$\text{Sprache} = \text{Syntax} + \text{Semantik} + \text{Pragmatik}$$

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

Prozess



UNI
FREIBURG



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

Inhalt der Vorlesung

Was ist Informatik?

Algorithmus

Computer

Algorithmen und Kochen

Beispiel

Eigenschaften

Programme und Programmiersprachen

Berechnungsprozess

Schluss

Exkurs:
Think like a computer scientist

- Der Ablauf eines Programms auf einem bestimmten Rechner zu einer bestimmten Zeit.
- In dieser Vorlesung spielt der Begriff des Prozesses keine große Rolle.
- In **Betriebssystemen** dreht sich alles um Prozesse. Z. B.: Wieviel Rechenzeit und wieviel Hauptspeicher auf welchem Prozessor bekommt welcher Prozess wann zugeteilt?

Ein-/Ausgabe, Algorithmus, Programm, (Berechnungs)prozess



- Ein **Algorithmus** ist eine Vorschrift zur Durchführung einer Berechnung.
- Ein bestimmtes **Ein-/Ausgabe-Verhalten** kann i.A. 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.
- Beliebig viele **Prozesse** können das gleiche Programm auf einem oder mehreren Computern auf der ganzen Welt ausführen.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Aphorismus



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer
Algorithmen und
Kochen

Beispiel

Eigenschaften
Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Ein Rechner ist ein Vollidiot mit Spezialbegabung. Er hat ein großes, präzises Gedächtnis und kann schneller rechnen als ein Mensch.

— Prof. Dr. Gerhard Goos (1962)

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

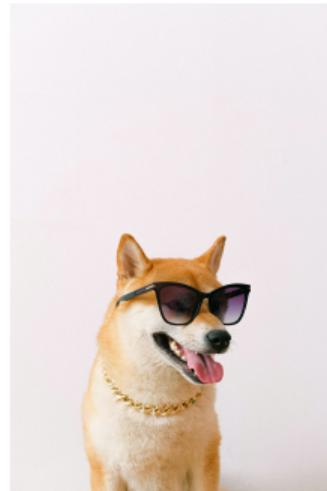
Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Exkurs: Think like a Dog



Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Computer

Algorithmen und
Kochen

Beispiel

Eigenschaften

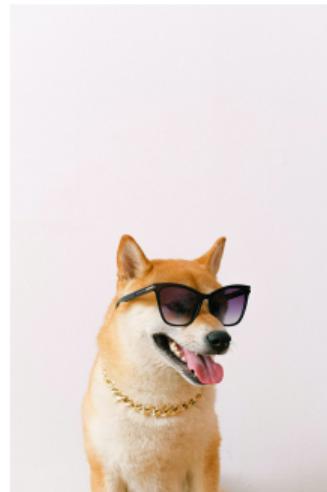
Programme und
Programmierspra-
chen

Berechnungspro-
zess

Schluss

Exkurs:
Think like a
computer
scientist

Exkurs: Think like a Dog





Exkurs: Think like a computer scientist

Denken wie ein Informatiker/eine Informatikerin



- Ein Studium vermittelt **Kenntnisse** und **Fähigkeiten**
- Es verändert aber auch die **Sicht auf die Welt**
- Informatiker tendieren dazu, in ihrer Umgebung nach dem **algorithmischen Kern** von Problemen zu suchen ...
- ...und diesen Kern dann auf dem Computer zu lösen

Beispielszenario: Die gemeinsame Reisekasse



- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Beispielszenario: Die gemeinsame Reisekasse



- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.
- Das ist einfacher ...

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Beispielszenario: Die gemeinsame Reisekasse



- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.
- Das ist einfacher ...
- ...und wird abwechselnd von den Gruppenmitgliedern übernommen.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Beispielszenario: Die gemeinsame Reisekasse



- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.
- Das ist einfacher ...
- ...und wird abwechselnd von den Gruppenmitgliedern übernommen.
- Zum Schluss haben die einen mehr bezahlt, die anderen weniger.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Beispielszenario: Die gemeinsame Reisekasse



- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.
- Das ist einfacher ...
- ...und wird abwechselnd von den Gruppenmitgliedern übernommen.
- Zum Schluss haben die einen mehr bezahlt, die anderen weniger.
- Es sind **Ausgleichszahlungen** nötig.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Beispielszenario: Die gemeinsame Reisekasse



Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

- Im Ausland gibt es oft nur eine gemeinschaftliche Rechnung, die einer in der Gruppe bezahlen muss.
- Das ist einfacher ...
- ... und wird abwechselnd von den Gruppenmitgliedern übernommen.
- Zum Schluss haben die einen mehr bezahlt, die anderen weniger.
- Es sind **Ausgleichszahlungen** nötig.
- Bei 3-4 Leuten einfach, bei 8-10 Leuten wird es unübersichtlich.

Beispiel



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



Beispiel



UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



Beispiel



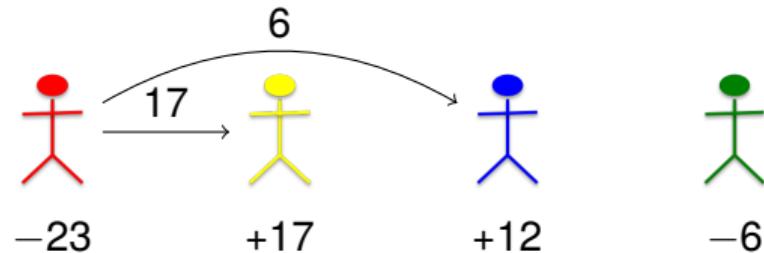
UNI
FREIBURG

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



Beispiel



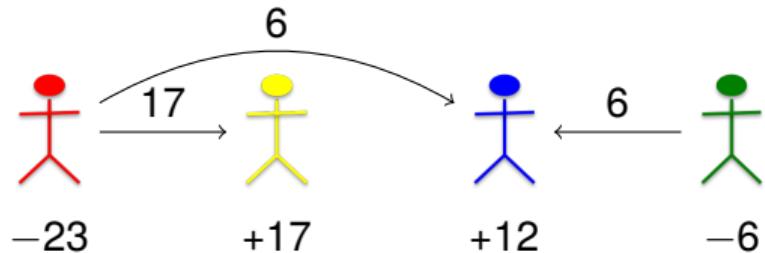
UNI
FREIBURG

Inhalt der
Vorlesung

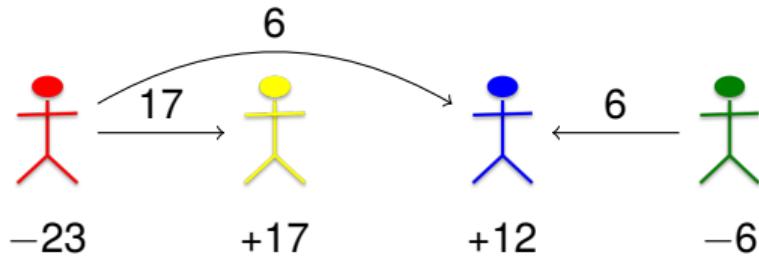
Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist



Beispiel



Nachteil: U.U. muss jemand mehrere Überweisungen tätigen oder jemand muss auf mehrere Überweisungen warten.

Das Reisekassen-Ausgleichszahlung-Problem (RAP)



Wir haben folgende **Anforderungen** für das RAP:

- Jeder sollte maximal eine Überweisung tätigen und eine Überweisung empfangen und ...
- ... der maximal zu überweisende Betrag sollte minimal sein.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Das Reisekassen-Ausgleichszahlung-Problem (RAP)



Wir haben folgende **Anforderungen** für das RAP:

- Jeder sollte maximal eine Überweisung tätigen und eine Überweisung empfangen und ...
- ... der maximal zu überweisende Betrag sollte minimal sein.

D.h. bestimme, wer wem was zu bezahlen hat:

- Finde eine Reihenfolge der Gruppenmitglieder, sodass die Zahlungen von links nach rechts getätigigt werden.
- Ansatz: Ausprobieren aller Reihenfolgen.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Einige mögliche Reihenfolgen



UNI
FREIBURG



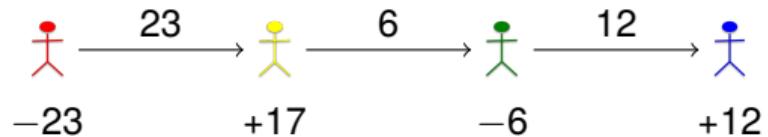
Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Einige mögliche Reihenfolgen



Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Einige mögliche Reihenfolgen



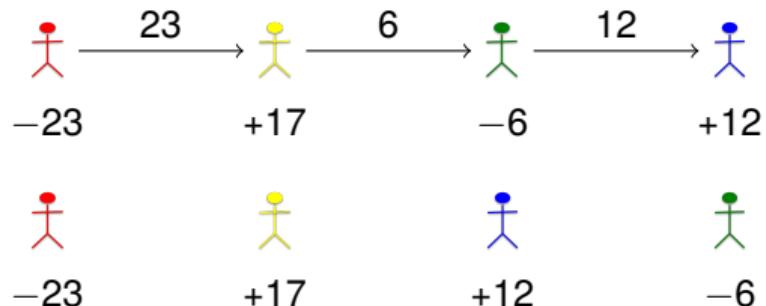
UNI
FREIBURG

Inhalt der
Vorlesung

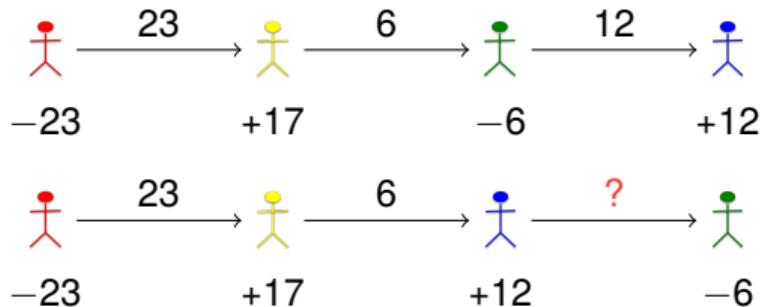
Was ist
Informatik?

Algorithmus

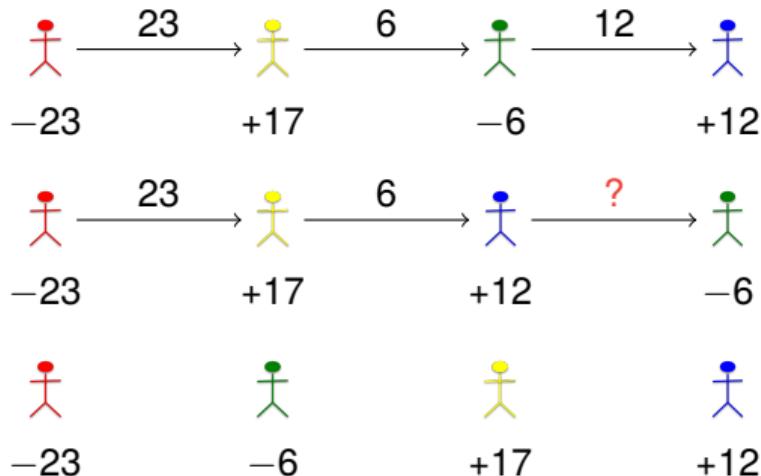
Exkurs:
Think like a
computer
scientist



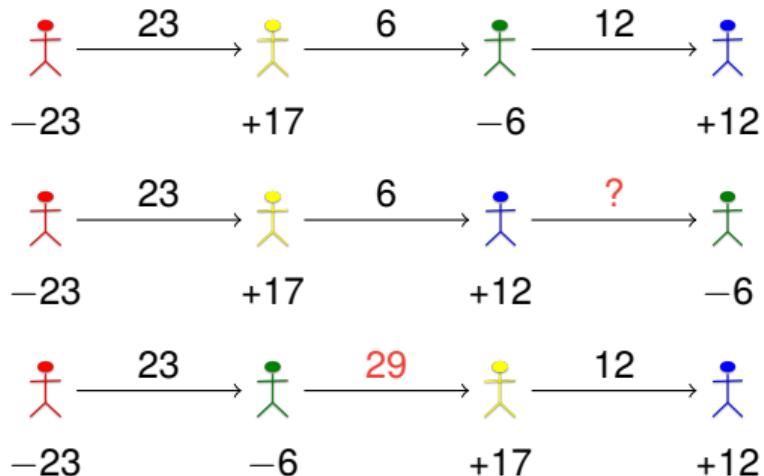
Einige mögliche Reihenfolgen



Einige mögliche Reihenfolgen



Einige mögliche Reihenfolgen



Reisekassen-Ausgleichszahlung-Algorithmus



UNI
FREIBURG

- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Reisekassen-Ausgleichszahlung-Algorithmus



- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,
 - 2 falls ja, bestimme und merke den Wert der maximalen Zahlung.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,
 - 2 falls ja, bestimme und merke den Wert der maximalen Zahlung.
- Gib eine Reihenfolge mit minimalem Wert zurück.

Inhalt der
Vorlesung

Was ist
Informatik?

Algorithmus

Exkurs:
Think like a
computer
scientist

Reisekassen-Ausgleichszahlung-Algorithmus



- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,
 - 2 falls ja, bestimme und merke den Wert der maximalen Zahlung.
- Gib eine Reihenfolge mit minimalem Wert zurück.
→ Es müssen $n! \approx \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$ Reihenfolgen betrachtet werden, was bei großem n sehr lange dauern kann...

Reisekassen-Ausgleichszahlung-Algorithmus



- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,
 - 2 falls ja, bestimme und merke den Wert der maximalen Zahlung.
- Gib eine Reihenfolge mit minimalem Wert zurück.
 - Es müssen $n! \approx \sqrt{2\pi n}(\frac{n}{e})^n$ Reihenfolgen betrachtet werden, was bei großem n sehr lange dauern kann...
 - Es gibt (vermutlich) keine effiziente Lösung, da das Problem **NP-schwer** ist.

Reisekassen-Ausgleichszahlung-Algorithmus



- Probiere systematisch alle Reihenfolgen (Permutationen) aus und führe für jede Reihenfolge folgendes durch:
 - 1 Prüfe ob nur positive Zahlungen erfolgen,
 - 2 falls ja, bestimme und merke den Wert der maximalen Zahlung.
- Gib eine Reihenfolge mit minimalem Wert zurück.
 - Es müssen $n! \approx \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$ Reihenfolgen betrachtet werden, was bei großem n sehr lange dauern kann...
 - Es gibt (vermutlich) keine effiziente Lösung, da das Problem **NP-schwer** ist.
- Der Urlaubs-Abschluss ist jedenfalls gerettet.