

# Informatik I

Tobias Joschko

DHBW Mannheim

WS 2021

# Algorithmen

# Algorithmen

# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

- ▶ endlich viele klare, wohldefinierte Schritte

# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

- ▶ endlich viele klare, wohldefinierte Schritte
- ▶ geht zurück auf arabischen Mathematiker **al-Chwarizmi**
  - ▶ **De numero Indorum**, erste Beschreibung des Rechnens mit Dezimalzahlen im 9. Jahrhundert

# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

- ▶ endlich viele klare, wohldefinierte Schritte
- ▶ geht zurück auf arabischen Mathematiker **al-Chwarizmi**
  - ▶ **De numero Indorum**, erste Beschreibung des Rechnens mit Dezimalzahlen im 9. Jahrhundert
- ▶ Beispiele für heutige Algorithmen:

# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

- ▶ endlich viele klare, wohldefinierte Schritte
- ▶ geht zurück auf arabischen Mathematiker **al-Chwarizmi**
  - ▶ **De numero Indorum**, erste Beschreibung des Rechnens mit Dezimalzahlen im 9. Jahrhundert
- ▶ Beispiele für heutige Algorithmen:
  - ▶ Computerprogramme
  - ▶ Rechenregeln z.B. bei Auszählungsverfahren für Wahlen
  - ▶ Kochrezepte



# Algorithmen

## Definition

Eindeutige Handlungsvorschrift zur Lösung eines Problems.

- ▶ endlich viele klare, wohldefinierte Schritte
- ▶ geht zurück auf arabischen Mathematiker **al-Chwarizmi**
  - ▶ **De numero Indorum**, erste Beschreibung des Rechnens mit Dezimalzahlen im 9. Jahrhundert
- ▶ Beispiele für heutige Algorithmen:
  - ▶ Computerprogramme
  - ▶ Rechenregeln z.B. bei Auszählungsverfahren für Wahlen
  - ▶ Kochrezepte
- ▶ Die Definition ist unscharf!
  - ▶ Was bedeutet *eindeutig* und *wohldefiniert*?

# Algorithmen

Formalisierung in den 30er Jahren durch Entwicklung abstrakter Maschinenmodelle

- ▶ Turingmaschine, Registermaschine
- ▶ Rekursion, While-Programme
- ▶ normale Familien von Algorithmen

Modelle haben minimalistische Beschreibungen

- ▶ formulieren Anforderungen an Programmiersprachen

# Algorithmen

Formalisierung in den 30er Jahren durch Entwicklung abstrakter Maschinenmodelle

- ▶ Turingmaschine, Registermaschine
- ▶ Rekursion, While-Programme
- ▶ normale Familien von Algorithmen

Modelle haben minimalistische Beschreibungen

- ▶ formulieren Anforderungen an Programmiersprachen
- ▶ deprimierendes Ergebnis: **Nicht jede mathematisch formulierbare Funktion kann auch berechnet werden**
  - ▶ z.B. Halteproblem

# Algorithmen

Formalisierung in den 30er Jahren durch Entwicklung abstrakter Maschinenmodelle

- ▶ Turingmaschine, Registermaschine
- ▶ Rekursion, While-Programme
- ▶ normale Familien von Algorithmen

Modelle haben minimalistische Beschreibungen

- ▶ formulieren Anforderungen an Programmiersprachen
- ▶ deprimierendes Ergebnis: **Nicht jede mathematisch formulierbare Funktion kann auch berechnet werden**
  - ▶ z.B. Halteproblem
- ▶ **Churchsche These**: Die intuitiv berechenbaren Funktionen sind genau die Turing-berechenbaren Funktionen.

# Algorithmen

## konkrete Beispiele für Programmierkonzepte

- ▶ Imperative Programmierung, Pseudocode
  - ▶ expliziter Kontrollfluss mit `if-then-else`, Schleifen etc.
  - ▶ z.B. C/C++, Java, Python
- ▶ Funktionale und deklarative Programmierung
  - ▶ festes Verfahren, mathematische Problembeschreibungen
  - ▶ z.B. Haskell
- ▶ endliche Automaten, Turingmaschinen, Excel

# Algorithmen

## Pseudocode

- ▶ Mischung aus natürlicher Sprache und mathematischer bzw. programmiersprachenartiger Notation.

# Algorithmen

## Pseudocode

- ▶ Mischung aus natürlicher Sprache und mathematischer bzw. programmiersprachenartiger Notation.
- ▶ Stellt Algorithmen **für den Menschen anschaulich** dar.
- ▶ Teile des Algorithmus können abgekürzt werden, wenn sie einfach sind oder an anderer Stelle beschrieben werden.

# Algorithmen

## Pseudocode

- ▶ Mischung aus natürlicher Sprache und mathematischer bzw. programmiersprachenartiger Notation.
- ▶ Stellt Algorithmen **für den Menschen anschaulich** dar.
- ▶ Teile des Algorithmus können abgekürzt werden, wenn sie einfach sind oder an anderer Stelle beschrieben werden.
- ▶ Notation in der Praxis oft ähnlich wie Pascal oder Python.
- ▶ Blöcke werden meist durch Einrückungen gekennzeichnet.



# Algorithmen

## Beispiel: Anleitung für die Informatik-Vorlesung

```
while Semester nicht zu Ende:
```

```
    Jede Woche an Vorlesung teilnehmen
```

```
    zuhören und mitreden
```

```
    if Inhalt unverständlich:
```

```
        Fragen stellen
```

```
Prüfung bestehen
```

# Algorithmen

## Beispiel: Anleitung für die Informatik-Vorlesung

```
while Semester nicht zu Ende:
    Jede Woche an Vorlesung teilnehmen
    zuhören und mitreden
    if Inhalt unverständlich:
        Fragen stellen
Prüfung bestehen
```

## Beispiel: Sortieren einer Liste von Zahlen (aufsteigend)

```
while liste nicht sortiert:
    finde x,y mit liste[x] > liste[y]
    vertausche l[x] und l[y]
```

- ▶ Wie findet man zwei solche Listenelemente?
- ▶ Wie vertauscht man zwei Listenelemente?

# Algorithmen

## Kontrollfluss

- ▶ Ausdrücke wie `if`, `for` und `while` heißen **Kontrollstrukturen**
- ▶ Sie steuern den **Kontrollfluss** des Programms/Algorithmus.

# Algorithmen

## Kontrollfluss

- ▶ Ausdrücke wie `if`, `for` und `while` heißen **Kontrollstrukturen**
- ▶ Sie steuern den **Kontrollfluss** des Programms/Algorithmus.
- ▶ Einfache Algorithmen bzw. ein grober Überblick kann auch durch **Flussdiagramme** gegeben werden (siehe Tafel).
- ▶ Flussdiagramme oder gut lesbarer Pseudocode können helfen, Ideen zu erklären und das eigene Programm zu strukturieren.

# Algorithmen

## Beispiele/Aufgaben

Formulieren Sie einen Algorithmus ...

1. ... der prüft, ob eine Liste eine bestimmte Zahl enthält.
2. ... für einen Primzahltest.
3. ... für ein schnelles Sortierverfahren.