

# Algorithmen und Datenstrukturen 1

Felix Ichtters, Lukas Dzielski\*

Sommersemester 2023

---

*Begleitmaterial zur Vorlesung 'Algorithmen und Datenstrukturen'.*

---

\*Universität Heidelberg

# Inhaltsverzeichnis

<b>1</b>	<b>Beweise</b>	<b>3</b>
1.1	Statements . . . . .	3
1.2	Formal mathematical proofs . . . . .	3
1.3	Set definition rule: Beispiel . . . . .	3
1.4	Macro-steps in proofs . . . . .	4
<b>2</b>	<b>Einführung</b>	<b>5</b>
2.1	... . . . .	5
<b>3</b>	<b>Fogen, Felder und Listen</b>	<b>6</b>
3.1	... . . . .	6
<b>4</b>	<b>Hashing</b>	<b>7</b>
4.1	... . . . .	7
<b>5</b>	<b>Sortieren</b>	<b>8</b>
5.1	... . . . .	8
<b>6</b>	<b>Prioritätslisten</b>	<b>9</b>
6.1	... . . . .	9
<b>7</b>	<b>Sortierte Listen</b>	<b>10</b>
7.1	... . . . .	10
<b>8</b>	<b>Graphenrepräsentation</b>	<b>11</b>
8.1	... . . . .	11
<b>9</b>	<b>Graphtraversierung</b>	<b>12</b>
9.1	... . . . .	12
<b>10</b>	<b>Kürteste Wege</b>	<b>13</b>
10.1	... . . . .	13
<b>11</b>	<b>Minimale Spannbäume</b>	<b>14</b>
11.1	... . . . .	14
<b>12</b>	<b>Optimierung</b>	<b>15</b>
12.1	... . . . .	15

# 1 Beweise

---

## 1.1 Statements

Ein **Statement/ Aussage** ist ein Mathematischer Ausdruck der entweder wahr oder Falsch ist.

**Beispiel:**

- $2 \in \{x \in \mathbb{R} | x < 5\}$  (wahr)
- $3^2 + 5^2 = 8^2$  (falsch)

Dabei werden Ausdrücke wie  $0 < x < 1$  verwendet um Mengen zu definieren.

$$A = \{x \in \mathbb{R} | 0 < x < 1\}$$

Wichtig ist hierbei der **Wahrheitswert** eines offenen Ausdrucks  $0 < x < 1$  hängt von gewähltem  $x$  ab. Also ist

- $x = 1/2$  (wahr)
- $x = 5$  (falsch)

Die **Domäne** ist hierfür wichtig zu beachten. Für  $\mathbb{N}$ , gibt es kein  $x$  s.t.  $0 < x < 1$ , aber es gibt welche für  $\mathbb{R}$

## 1.2 Formal mathematical proofs

Ein **formaler mathematischer Beweis** besteht aus einer nummerierten Sequent von **wahren Aussagen**. Jede Aussage in einem Beweis ist ein **Annahme** oder **folgt** aus vorherigen Aussagen durch Ableitungsregel/ Inferenzregel (rule of inference). Die Letzte Aussage ist die die wir bewiesen haben.

$\Rightarrow$  Offene Aussagen können in Beweisen nicht auftreten!

**Beispiel einer Inferenzregel: set definition rule**

Wenn ein Element in einer Menge ist, dann können wir definierende Eigenschaften ableiten. Andererseits, wenn es die definierende Eigenschaften erfüllt, dann können wir ableiten das das Element in der Menge ist.

## 1.3 Set definition rule: Beispiel

Definiere  $C = \{x \in \mathbb{R} | x < 2\}$

( $x < 2 \wedge x \in \mathbb{R}$  ist die definierende Eigenschaft). Dabei gibt es zwei Möglichkeiten für die Ableitung

- Möglichkeit 1
1.  $a \in C$
  2.  $a < 2 \wedge a \in \mathbb{R} (1; def C)$

- Möglichkeit 2
1.  $b < 2 \wedge b \in \mathbb{R}$
  2.  $b \in C(1; def C)$

Jede Aussage in dem Bewies hat eine Nummer. Wir begründen wie wir eine Aussage ableiten, zb  $(1; def C)$  bedeutet wir leiten die aktuelle Aussage aus Aussage 1 mit der Definition von  $C$  und der set definition rule ab.

**Bemerkung:**  $\wedge b \in R$

wird oft ausgelassen, wenn Kontext es zulässt.

## 1.4 Macro-steps in proofs

**Problem:**

Schauen wir uns folgenden Bewwis an:

(ass = Annahme (assumption) und prop = Eigenschaft(property))

*Beweis.* **Annahme:**

1.  $X = \{x \in \mathbb{R} \mid x < 1\}$
2.  $a \in X$

**Zeige:**  $a < 2$

- |    |           |                   |
|----|-----------|-------------------|
| 1. | $a \in X$ | (ass 2)           |
| 2. | $a < 1$   | (1, ass 1; def X) |
| 3. | $1 < 2$   | (prop R)          |
| 4. | $a < 2$   | (2, 3; prop R)    |

□

**Ist das ein akzrptabler Beweis?** Akzeptanz von Makro-Schritten wie "**prop**  $\mathbb{R}$ " hängt von der Zielgruppe ab!

Welche Eigenschaft von  $\mathbb{R}$  wurde benutzt?

## 2 Einführung

---

### 2.1 ...

...

## 3 Fogen, Felder und Listen

---

### 3.1 ...

...

## 4 Hashing

---

### 4.1 ...

...

## 5 Sortieren

---

### 5.1 ...

...



## 6 Prioritätslisten

---

### 6.1 ...

...

## 7 Sortierte Listen

---

### 7.1 ...

...

## 8 Graphenrepräsentation

---

### 8.1 ...

...

## 9 Graphtravesierung

---

### 9.1 ...

...

## 10 Kürteste Wege

---

### 10.1 ...

...

## 11 Minimale Spannbäume

---

### 11.1 ...

...

## 12 Optimierung

---

### 12.1 ...

...