

# Grundbegriffe der Informatik

## Tutorium 33

Lukas Bach, [lukas.bach@student.kit.edu](mailto:lukas.bach@student.kit.edu) | 19.01.2017



Lukas Bach, lu-  
kas.bach@student.kit.edu

## Wie stellen wir Graphen da?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Repräsentation von Graphen

Wie stellen wir Graphen da?

Adjazenzlisten

Erreichbarkeit

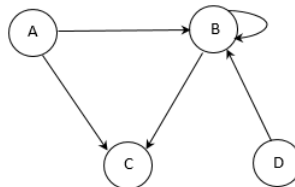
Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation



Anschaulich ja, aber wie können wir Graphen z.B. mit Java realisieren?

# Objektorientierte Repräsentation von Graphen

## Klassenmodell?

### Repräsentation von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Objektorientierte Repräsentation von Graphen

## Klassenmodell?

```
class Vertex {  
    String name; //Genauer Inhalt interessiert uns nicht  
}
```

```
class Edge {  
    Vertex start;  
    Vertex end;  
}
```

```
class Graph {  
    Vertex[] vertices;  
    Edge[] edges;  
}
```

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

## Repräsentation von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

## Repräsentation von Graphen

Adjazenzlisten

Erreichbarkeit

**+ Intuitiv**

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Repräsentation von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

+ Intuitiv

– Es lassen sich nur schwer Algorithmen hierfür entwerfen (z.B. gilt  
 $(x, y) \in E?$ )



Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Jeder Knoten speichert seine Nachbarn:

```
class Vertex {  
    String name; //Genauer Inhalt interessiert uns nicht  
    Vertex[] neighbours; //Alle Nachbarknoten  
}  
  
class Graph {  
    Vertex[] vertices;  
    Edge[] edges;  
}
```

- + Speicherplatzeffizient bei wenigen Kanten im Vergleich zur Knotenanzahl ( $|E| \ll |V|^2$ )
- + Flexibel mit verketteten Listen statt Arrays (Leichtes Hinzufügen und Entfernen)

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

**Adjazenzlisten**

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

## ■ Was ist eine Adjazenzmatrix?

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Repräsentation von Graphen

### Adjazenzlisten

### Erreichbarkeit

### Zwei-Erreichbarkeit

### Erreichbarkeit

### Algorithmus

## Komplexitätstheorie

### O-Notation

- Was ist eine Adjazenzmatrix?
- Zu allen Paaren  $(i, j)$  mit  $i, j \in V$  wird gespeichert, ob  $(i, j) \in E$  gilt

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Was ist eine Adjazenzmatrix?
- Zu allen Paaren  $(i, j)$  mit  $i, j \in V$  wird gespeichert, ob  $(i, j) \in E$  gilt
- Zweidimensionales Array

```
class Graph {  
    boolean[] [] edges; //Größe  $|V| \times |V|$   
}
```

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

**Adjazenzlisten**

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Repräsentation von Graphen

Adjazenzlisten

**+** Speicherplatzeffizient bei annähernd maximaler Anzahl von Kanten  
( $|E| \approx |V|^2$ )

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation



## Repräsentation von Graphen

Adjazenzlisten

+ Speicherplatzeffizient bei annähernd maximaler Anzahl von Kanten  
( $|E| \approx |V|^2$ )

Erreichbarkeit

+ Algorithmen aus linearer Algebra können verwendet werden  
(Matrizenrechnung)

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Repräsentation von Graphen

Adjazenzlisten

+ Speicherplatzeffizient bei annähernd maximaler Anzahl von Kanten  
( $|E| \approx |V|^2$ )

Erreichbarkeit

+ Algorithmen aus linearer Algebra können verwendet werden  
(Matrizenrechnung)

Zwei-Erreichbarkeit

Erreichbarkeit

– nicht flexibel

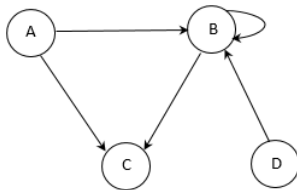
Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Gebe alle Adjazenlisten und die Adjazenzmatrix für diesen Graphen an:



Repräsentation  
von Graphen

Adjazenlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

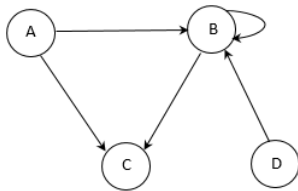
Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Gebe alle Adjazenlisten und die Adjazenzmatrix für diesen Graphen an:



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Repräsentation  
von Graphen

Adjazenlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Repräsentation von zweistelligen Relationen durch Matrizen

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wir können jede endliche zweistellige Relation durch eine Matrix darstellen!

## Aufgabe

Stelle die Kleiner-Gleich-Relation auf der Menge  $\{0, 1, 2, 3\}$  dar!

# Repräsentation von zweistelligen Relationen durch Matrizen

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wir können jede endliche zweistellige Relation durch eine Matrix darstellen!

## Aufgabe

Stelle die Kleiner-Gleich-Relation auf der Menge  $\{0, 1, 2, 3\}$  dar!

$$R_{\leq} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

**Erreichbarkeit**

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

## ■ Algorithmisches Problem

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation



Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

## Wege-Problem

Gegeben einem Graphen  $G = (V, E)$ . Ist für  $i, j \in V$  auch  $(i, j) \in E^*$ ?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

## Wege-Problem

Gegeben einem Graphen  $G = (V, E)$ . Ist für  $i, j \in V$  auch  $(i, j) \in E^*$ ?

### Ziel

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

## Wege-Problem

Gegeben einem Graphen  $G = (V, E)$ . Ist für  $i, j \in V$  auch  $(i, j) \in E^*$ ?

### Ziel

- Gegeben: Adjazenzmatrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

## Wege-Problem

Gegeben einem Graphen  $G = (V, E)$ . Ist für  $i, j \in V$  auch  $(i, j) \in E^*$ ?

### Ziel

- Gegeben: Adjazenzmatrix
- Gesucht: Zugehörige **Wegematrix**, für die gilt:

- Algorithmisches Problem
- Intuitiv: Gibt es einen Weg von  $i$  nach  $j$ ?

## Wege-Problem

Gegeben einem Graphen  $G = (V, E)$ . Ist für  $i, j \in V$  auch  $(i, j) \in E^*$ ?

### Ziel

- Gegeben: Adjazenzmatrix
- Gesucht: Zugehörige **Wegematrix**, für die gilt:

$$W_{ij} = \begin{cases} 1 & , \text{ falls ein Weg von } i \text{ nach } j \text{ existiert} \\ 0 & , \text{ sonst} \end{cases}$$

Lukas Bach, lu-  
kas.bach@student.kit.edu

Was wisst ihr zu folgenden Begriffen?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

Was wisst ihr zu folgenden Begriffen?

■ Matrizenmultiplikation

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation



Lukas Bach, lu-  
kas.bach@student.kit.edu

Was wisst ihr zu folgenden Begriffen?

- Matrizenmultiplikation
- Matrizenaddition

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Was wisst ihr zu folgenden Begriffen?

- Matrizenmultiplikation
- Matrizenaddition
- Potenzieren

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Was wisst ihr zu folgenden Begriffen?

- Matrizenmultiplikation
- Matrizenaddition
- Potenzieren
- Einheitsmatrix

## Was wisst ihr zu folgenden Begriffen?

- Matrizenmultiplikation
- Matrizenaddition
- Potenzieren
- Einheitsmatrix
- Nullmatrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Quadriere die Adjazenzmatrix von vorhin:  $A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$

# Quadrierte Adjazenzmatrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

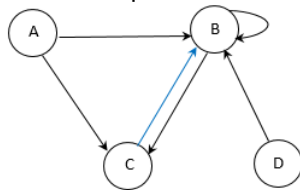
Quadriere die Adjazenzmatrix von vorhin:  $A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$

## Ergebnis

$$A^2 = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

## Aufgabe

Bilde und quadriere die Adjazenzmatrix des veränderten Graphen:



Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

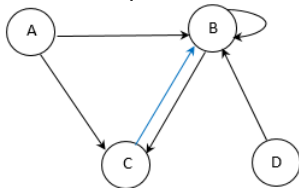
Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Bilde und quadriere die Adjazenzmatrix des veränderten Graphen:



$$A' = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \text{ und } A'^2 = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$



## Aufgabe

Was fällt euch auf? Wann steht in  $A'^2$  eine 1, wann eine 2 und was bedeutet das für unseren Graphen?

$$A' = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad A'^2 = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

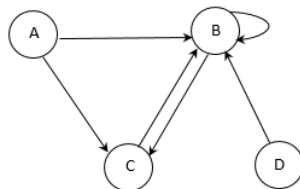
Komplexitätstheorie

O-Notation

## Aufgabe

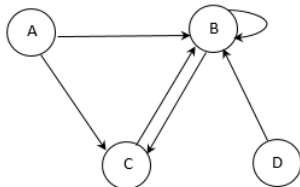
Was fällt euch auf? Wann steht in  $A'^2$  eine 1, wann eine 2 und was bedeutet das für unseren Graphen?

$$A' = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad A'^2 = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$



## Aufgabe

Was fällt euch auf? Wann steht in  $A'^2$  eine 1, wann eine 2 und was bedeutet das für unseren Graphen?



$$A' = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad A'^2 = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

	$b_{11}$	$b_{12}$	$b_{13}$		
	$b_{21}$	$b_{22}$	$b_{23}$		
	$b_{31}$	$b_{32}$	$b_{33}$		
$a_{11}$	$a_{12}$	$a_{13}$	$c_{11}$	$c_{12}$	$c_{31}$
$a_{21}$	$a_{22}$	$a_{23}$	$c_{21}$	$c_{22}$	$c_{23}$
$a_{31}$	$a_{32}$	$a_{33}$	$c_{31}$	$c_{32}$	$c_{33}$

**Tipp:**  $c_{11} = a_{11} \cdot b_{11} + a_{12} \cdot b_{21} + a_{13} \cdot b_{31}$

## Lösung

In der  $i$ -ten Zeile und  $j$ -ten Spalte von  $A^2$  steht die Anzahl der Wege von  $i$  nach  $j$  der Länge zwei.

→  $(A^2)_{ij}$  = Anzahl der Pfade von  $i$  nach  $j$  der Länge zwei.

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Lösung

In der  $i$ -ten Zeile und  $j$ -ten Spalte von  $A^2$  steht die Anzahl der Wege von  $i$  nach  $j$  der Länge zwei.

→  $(A^2)_{ij}$  = Anzahl der Pfade von  $i$  nach  $j$  der Länge zwei.

## Aufgabe

Habt ihr Ideen, wie man herausfindet, zwischen welchen Knoten Pfade der Länge  $n$  existieren?

## Lösung

In der  $i$ -ten Zeile und  $j$ -ten Spalte von  $A^2$  steht die Anzahl der Wege von  $i$  nach  $j$  der Länge zwei.

→  $(A^2)_{ij}$  = Anzahl der Pfade von  $i$  nach  $j$  der Länge zwei.

## Aufgabe

Habt ihr Ideen, wie man herausfindet, zwischen welchen Knoten Pfade der Länge  $n$  existieren?

## Lösung

Betrachte  $A^n$ !

Lukas Bach, lu-  
kas.bach@student.kit.edu

Eigentlich interessiert uns nur, ob ein Pfad der Länge zwei existiert und nicht wie viele...

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

**Zwei-Erreichbarkeit**

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Eigentlich interessiert uns nur, ob ein Pfad der Länge zwei existiert und nicht wie viele...

## Definition Signum-Funktion

$$\text{sgn} : \mathbb{R} \rightarrow \mathbb{R}$$

$$x \mapsto \begin{cases} 1 & , \text{ falls } x > 0 \\ 0 & , \text{ falls } x = 0 \\ -1 & , \text{ falls } x < 0 \end{cases}$$



Eigentlich interessiert uns nur, ob ein Pfad der Länge zwei existiert und nicht wie viele...

## Definition Signum-Funktion

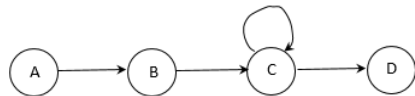
$$\text{sgn} : \mathbb{R} \rightarrow \mathbb{R}$$

$$x \mapsto \begin{cases} 1 & , \text{ falls } x > 0 \\ 0 & , \text{ falls } x = 0 \\ -1 & , \text{ falls } x < 0 \end{cases}$$

$\text{sgn}(A^2)$  liefert uns die Zwei-Erreichbarkeitsmatrix

## Aufgabe

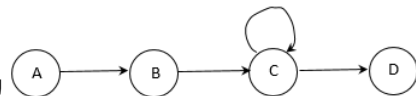
Gebe  $A^0$ ,  $A^2$  und die Wegematrix  $W$  an!



## Aufgabe

Gebe  $A^0$ ,  $A^2$  und die Wegematrix  $W$  an!

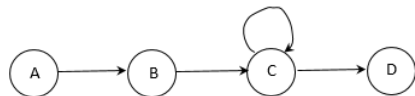
$$A^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



## Aufgabe

Gebe  $A^0$ ,  $A^2$  und die Wegematrix  $W$  an!

$$A^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad A^2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$



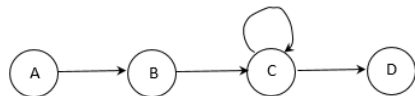
## Aufgabe

Gebe  $A^0$ ,  $A^2$  und die Wegematrix  $W$  an!

$$A^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Für Pfade beliebiger Länge erhalten wir:

$$W = \text{sgn}(A^0 + A^1 + A^2 + A^3 + \dots) = \text{sgn}\left(\sum_{i=0}^{\infty} A^i\right)$$

Für Pfade beliebiger Länge erhalten wir:

$$W = \text{sgn}(A^0 + A^1 + A^2 + A^3 + \dots) = \text{sgn}\left(\sum_{i=0}^{\infty} A^i\right)$$

Wir können nicht unendlich lange addieren... Ist das ein **Problem**?

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

**Erreichbarkeit**

Algorithmus

Komplexitätstheorie

O-Notation



Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, muss mindestens ein Knoten doppelt vorgekommen sein!

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, muss mindestens ein Knoten doppelt vorgekommen sein! Der Pfad  $p$  enthält also einen Zyklus, den wir raus kürzen können.

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Erreichbarkeit- unendlich addieren?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, muss mindestens ein Knoten doppelt vorgekommen sein! Der Pfad  $p$  enthält also einen Zyklus, den wir raus kürzen können.

## Ergebnis

Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, existiert auch ein Pfad  $p'$  der Länge  $< n$ .

# Erreichbarkeit- unendlich addieren?

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, muss mindestens ein Knoten doppelt vorgekommen sein! Der Pfad  $p$  enthält also einen Zyklus, den wir raus kürzen können.

## Ergebnis

Wenn ein Pfad  $p$  der Länge  $\geq n := |V|$  zwischen  $i \neq j$  existiert, existiert auch ein Pfad  $p'$  der Länge  $< n$ .

Für Pfade beliebiger Länge erhalten wir:

$$W = \text{sgn}(A^0 + A^1 + A^2 + A^3 + \dots) = \text{sgn}\left(\sum_{i=0}^{\infty} A^i\right) = \text{sgn}\left(\sum_{i=0}^{n-1} A^i\right)$$

# Einfacher Algorithmus zu Berechnung der Wegematrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

*⟨Matrix A sei die Adjazenzmatrix⟩*

$W \leftarrow 0$

**for**  $i \leftarrow 0$  **to**  $n - 1$  **do**

$M \leftarrow I$

**for**  $j \leftarrow 1$  **to**  $i$  **do**

$M \leftarrow M \cdot A$

**od**

$W \leftarrow W + M$

**od**

$W \leftarrow \text{sgn}(W)$

# Einfacher Algorithmus zu Berechnung der Wegematrix

*⟨Matrix A sei die Adjazenzmatrix⟩*

$W \leftarrow 0$

**for**  $i \leftarrow 0$  **to**  $n - 1$  **do**

$M \leftarrow I$

**for**  $j \leftarrow 1$  **to**  $i$  **do**

$M \leftarrow M \cdot A$

**od**

    {  $M = A^i$  }

$W \leftarrow W + M$

    {  $W = \sum_{k=0}^i A^k$  }

**od**

$W \leftarrow \text{sgn}(W)$

{  $W$  ist die Wegematrix }

Aufwand:

$$\begin{aligned} & \left( \sum_{i=0}^{n-1} i \right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2 \\ &= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2 \end{aligned}$$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Einfacher Algorithmus zu Berechnung der Wegematrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

*⟨Matrix A sei die Adjazenzmatrix⟩*

$W \leftarrow 0$

**for**  $i \leftarrow 0$  **to**  $n - 1$  **do**

$M \leftarrow 1$

**for**  $j \leftarrow 1$  **to**  $i$  **do**

$M \leftarrow M \cdot A$

**od**

$\{ M = A^i \}$

$W \leftarrow W + M$

$\{ W = \sum_{k=0}^i A^k \}$

**od**

$W \leftarrow \text{sgn}(W)$

$\{ W \text{ ist die Wegematrix} \}$

Aufwand:

$$\begin{aligned} & \left( \sum_{i=0}^{n-1} i \right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2 \\ &= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2 \end{aligned}$$

Wie könnte man diesen Algorithmus schneller machen?

# Einfacher Algorithmus zu Berechnung der Wegematrix

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

```
W ← 0
M ← I
for i ← 0 to n - 1 do
    W ← W + M
    M ← M · A
od
W ← sgn(W)
```

für  $A^i$  kann man  $A^{i-1}$  wiederverwenden

Aufwand:

$$n \cdot (n^2 + (2n^3 - n^2)) + n^2 = 2n^4 + n^2$$



Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

**Komplexitätstheorie**

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

## Wichtige Komplexitätsmaße:

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

## Wichtige Komplexitätsmaße:

- Speicherplatzbedarf

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wichtige Komplexitätsmaße:

- Speicherplatzbedarf
- Rechen- bzw. Laufzeit

Unterscheidung in

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wichtige Komplexitätsmaße:

- Speicherplatzbedarf
- Rechen- bzw. Laufzeit

Unterscheidung in

- Best Case (oft uninteressant)

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wichtige Komplexitätsmaße:

- Speicherplatzbedarf
- Rechen- bzw. Laufzeit

Unterscheidung in

- Best Case (oft uninteressant)
- Average Case (schwierig zu finden, deswegen selten angegeben)

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Wichtige Komplexitätsmaße:

- Speicherplatzbedarf
- Rechen- bzw. Laufzeit

Unterscheidung in

- Best Case (oft uninteressant)
- Average Case (schwierig zu finden, deswegen selten angegeben)
- Worst Case (meistens angegeben)

## Definition

Seien  $g, f : \mathbb{N}_0 \rightarrow \mathbb{R}_0^+$  Funktionen. Dann wächst  $g$  asymptotisch genauso schnell wie  $f$  genau dann, wenn gilt:

$$\exists c, c' \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : c \cdot f(n) \leq g(n) \leq c' \cdot f(n)$$



## Definition

Seien  $g, f : \mathbb{N}_0 \rightarrow \mathbb{R}_0^+$  Funktionen. Dann wächst  $g$  asymptotisch genauso schnell wie  $f$  genau dann, wenn gilt:

$$\exists c, c' \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : c \cdot f(n) \leq g(n) \leq c' \cdot f(n)$$

## Notation

$f \asymp g$  oder  $f(n) \asymp g(n)$  ("asymptotisch gleich")

## Bemerkung

$\asymp$  ist eine Äquivalenzrelation

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Definition

$$\Theta(f) = \{g \mid g \asymp f\}$$

## Definition

$$\Theta(f) = \{g \mid g \asymp f\}$$

## Satz

$$\forall a, b \in \mathbb{R}_+ : \Theta(a \cdot f) = \Theta(b \cdot f)$$

## Obere Schranke (Worst-Case Approximation)

$$O(f) = \{g \mid \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \leq c \cdot f(n)\}$$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Obere und untere Schranke

## Obere Schranke (Worst-Case Approximation)

$$O(f) = \{g | \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \leq c \cdot f(n)\}$$

## Untere Schranke (Best-Case Approximation)

$$\Omega(f) = \{g | \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \geq c \cdot f(n)\}$$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Obere Schranke (Worst-Case Approximation)

$$O(f) = \{g | \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \leq c \cdot f(n)\}$$

## Untere Schranke (Best-Case Approximation)

$$\Omega(f) = \{g | \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \geq c \cdot f(n)\}$$

## Notation

- $g \preceq f$  falls  $g \in O(f)$  bzw.  $g$  wächst asymptotisch höchstens so schnell wie  $f$
- $g \succeq f$  falls  $g \in \Omega(f)$  bzw.  $g$  wächst asymptotisch mindestens so schnell wie  $f$

## Obere Schranke (Worst-Case Approximation)

$$O(f) = \{g \mid \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \leq c \cdot f(n)\}$$

## Untere Schranke (Best-Case Approximation)

$$\Omega(f) = \{g \mid \exists c \in \mathbb{R}_+ : \exists n_0 \in \mathbb{N}_0 : \forall n \geq n_0 : g(n) \geq c \cdot f(n)\}$$

## Notation

- $g \preceq f$  falls  $g \in O(f)$  bzw.  $g$  wächst asymptotisch höchstens so schnell wie  $f$
- $g \succeq f$  falls  $g \in \Omega(f)$  bzw.  $g$  wächst asymptotisch mindestens so schnell wie  $f$

## Bemerkung

Es gilt  $\Theta(f) = O(f) \cap \Omega(f)$

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Lemma

$$\log_a n \in \Theta(\log_b n)$$

## Beispiel

$$\log_2 n \in \Theta(\log_8 n)$$



Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Lemma

$$\log_a n \in \Theta(\log_b n)$$

### Beispiel

$$\log_2 n \in \Theta(\log_8 n)$$

### Beweis

$$\frac{1}{3} \log_2 n$$

## Lemma

$$\log_a n \in \Theta(\log_b n)$$

### Beispiel

$$\log_2 n \in \Theta(\log_8 n)$$

### Beweis

$$\frac{1}{3} \log_2 n = \frac{1}{\log_2 8} \log_2 n$$

## Lemma

$$\log_a n \in \Theta(\log_b n)$$

### Beispiel

$$\log_2 n \in \Theta(\log_8 n)$$

### Beweis

$$\frac{1}{3} \log_2 n = \frac{1}{\log_2 8} \log_2 n = \frac{\log_2 n}{\log_2 8}$$

## Lemma

$$\log_a n \in \Theta(\log_b n)$$

### Beispiel

$$\log_2 n \in \Theta(\log_8 n)$$

### Beweis

$$\frac{1}{3} \log_2 n = \frac{1}{\log_2 8} \log_2 n = \frac{\log_2 n}{\log_2 8} = \log_8 n \leq \log_2 n$$

# Grundbegriffe der Informatik

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Gilt  $\log_2(n^{20}) \in \Theta(\log n)$

# Grundbegriffe der Informatik

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

## Aufgabe

Gilt  $\log_2(n^{20}) \in \Theta(\log n)$

## Lösung

Ja, denn  $\log_2(n^{20}) = 20 \cdot \log_2 n$

# Grundbegriffe der Informatik

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

## Probeklausur

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation

# Grundbegriffe der Informatik

Lukas Bach, lu-  
kas.bach@student.kit.edu

Repräsentation  
von Graphen

Adjazenzlisten

Erreichbarkeit

Zwei-Erreichbarkeit

Erreichbarkeit

Algorithmus

Komplexitätstheorie

O-Notation



*That's all Folks!*