

X. Hausaufgabe im Modul „Berechenbarkeit & Komplexität“

Gruppe XYZ

1 Grundlegendes

Einzelne Zeilenumbrüche sind Latex egal. Mit einem doppelten Umbruch erzeugt man einen neuen Absatz. Der Übersicht halber sollte man die Zeilen im Latex-Quellcode trotzdem nicht zu lang machen.

Mit `\newcommand` lassen sich eigene Befehle definieren: Jede supercalifragilisticexpialgetische Instanz lässt sich mit einem supercalifragilisticexpialgetischen Algorithmus in supercalifragilisticexpialgetischer Zeit lösen. (Sehr nützlich, wenn man später Dinge umbenennen will.)

2 Formatierung

Ich bin kursiv. **Ich bin fett.** ICH BIN IN KAPITÄLCHEN (für Problemnamen). „Ich stehe in Anführungszeichen.“

Hier kommt eine Formel: $a = b + c$. Die nächste Formel kriegt mehr Platz und wird am $\&$ -Zeichen ausgerichtet:¹

$$\begin{aligned} f : \mathbb{R} &\rightarrow \mathbb{R} \\ x &\mapsto x^2 \end{aligned}$$

Ohne Sternchen gibt's Zeilennummern:

$$a = 14^2 + 17^2 \tag{1}$$

$$\sqrt{17} > 1 \tag{2}$$

Hoch- und tiefgestellt wird so: $x_1, x^2, x_{i,j}^t$

Brüche schreibt man $\frac{n!}{k!(n-k)!}$ oder im Fließtext auch einfach $n!/(k!(n-k)!)$.

¹Man kann auch an mehreren Stellen ausrichten.

Ein paar weitere Operatoren:

$$\sum_{i=1}^n \int_a^b \lim_{x \rightarrow \infty} \max_{y \geq x} \left\{ x \cdot y \mid [x^2 - y^2] \subseteq \tilde{\Omega} \right\}$$

Wer den Namen für ein bestimmtes Symbol sucht schaut am besten auf <http://detexify.kirelabs.org>

Wenn man Text innerhalb einer Formel einfügen möchte, muss dieser mit `\text` ausgezeichnet werden:

$$V = \frac{4}{3}\pi z^3 \quad \text{Volumen einer Kugel mit Radius } z$$

$$V = \pi z^2 a = \pi z z a \quad \text{Volumen einer Pizza mit Radius } z \text{ und Dicke } a$$

Versuche Zeilenumbrüche an mathematischen Ausdrücken wie z. B. dem G hier zu vermeiden. Verwende die Tilde im Latex-Code, um das zu vermeiden, wie folgt.

Versuche Zeilenumbrüche an mathematischen Ausdrücken wie z. B. dem G hier zu vermeiden.

Und noch was: Sätze sollten auch nicht mit mathematischen Ausdrücken beginnen. Also, statt „ G ist ein Baum.“ besser „Der Graph G ist ein Baum.“.

3 Theoreme & Co

Theorem 3.1. VERTEX COVER ist lösbar in exponentieller Zeit.

Beweis. Wir geben später Algorithmus 1 für VERTEX COVER an. Dieser läuft offensichtlich in exponentieller Zeit. (Dies ist kein echter Beweis!) \square

Korollar 3.2. VERTEX COVER ist entscheidbar.

Beispiel 1. Ein interessantes Beispiel.

Anmerkung. Eine Anmerkung.

4 Abbildungen

Beispiele für Graphen sind in Abbildung 1 abgebildet. Hierbei ist Abbildung 1b ein *gerichteter Graph* (auch *Digraph*).

Wer zu faul ist, die Bilder in tikz zu erstellen, kann sich auch <http://ipe.otfried.org/> anschauen.

Tabelle 1 ist eine große Tabelle. Normalerweise versucht Latex selbst, einen geeigneten Ort für Abbildungen zu finden.

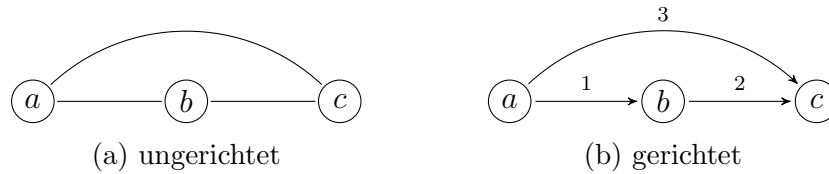


Abbildung 1: Beispiele für Graphen. Achtung: *label* immer **nach** *caption*, sonst gehen die Verweise kaputt.

Tabelle 1: Ergebnisse der Laufzeitexperimente. Alle Zeiten sind in Sekunden angegeben. Im Gegensatz zu Abbildungen sind Tabellenüberschriften über der Tabelle zu platzieren.

	file	n	m	WCC [s]	HKN [s]	CHLS [s]
Clustering	email	1K	5K	0.67	48.75	2.51
	hep-th	7K	15K	0.43	1.35	3.75
	netscience	1K	2K	0.24	0.37	0.07
	PGPgiantcompo	10K	24K	0.67	1.70	3.51
Co-author	citationCiteseer	268K	1.1M	21.46	83.90	—
	coAuthorsCiteseer	227K	0.8M	11.46	50.35	—
	coAuthorsDBLP	289K	0.9M	15.76	78.51	—
	coPapersCiteseer	434K	16M	—	—	—

5 Pseudocode

Pseudocode kann manchmal hilfreich zur besseren Darstellung des Algorithmus sein. Allerdings kann man sich auch schnell darin verlieren. Wenn der Pseudocode praktisch eine leicht angepasste Kopie von echtem Programm-Code (z.B. Python) ist, dann ist er sehr schwer verständlich und oft nicht hilfreich. Insbesondere, wenn Ihr eine Aufgabe mit dynamischer Programmierung löst, kann Pseudocode hinderlich sein.

Data : Ein Graph $G = (V, E)$.

Result : Größe eines kleinsten Vertex Covers.

```

1 for  $k = 0$  to  $|V|$  do
2   foreach  $V' \subseteq V$  with  $|V'| = k$  do
3      $sol \leftarrow \text{true}$ ;
4     foreach  $\{u, v\} \in E$  do
5       if  $u \notin V' \wedge v \notin V'$  then
6          $sol \leftarrow \text{false}$ ;
7       end
8     end
9     if  $sol = \text{true}$  then
10      return  $k$ 
11    end
12  end
13 end

```

Algorithmus 1 : TestVC - Ein Beispielalgorithmus für Vertex Cover.

Der folgende Algorithmus Algorithmus 2 läuft zum Beispiel in $O(\log k)$ Zeit.

```

1 while  $k \neq 0$  do
2   if  $k = 1$  then
3      $k = 0$ 
4   else
5      $k = \lceil \frac{k}{2} \rceil$ 
6   end
7 end

```

Algorithmus 2 : Ein Beispielalgorithmus mit einer WHILE-Schleife und einem IF-THEN-ELSE-Konstrukt.

6 Automaten und Turingmaschinen

Nachfolgend der DFA von Folie 8 und die TM von Folie 14:

$M = (\{z_0, z_1, z_2\}, \{0, 1\}, \delta, z_0, \{z_2\})$ mit

δ	z_0	z_1	z_2
0	z_0	z_2	z_1
1	z_1	z_0	z_2

oder

$M = (\{z_0, z_1, z_2, z_e\}, \Sigma = \{0, 1\}, \Gamma = \{0, 1, \square\}, \delta, z_0, \square, \{z_e\})$ mit

δ	0	1	\square
z_0	$(z_0, 0, R)$	$(z_0, 1, R)$	(z_1, \square, L)
z_1	$(z_2, 1, L)$	$(z_1, 0, L)$	$(z_e, 1, N)$
z_2	$(z_2, 0, L)$	$(z_2, 1, L)$	(z_e, \square, R)

Wer will, kann auch die Zustandsgraphen angeben (auch wenn wir empfehlen, diese per Hand zu malen):

