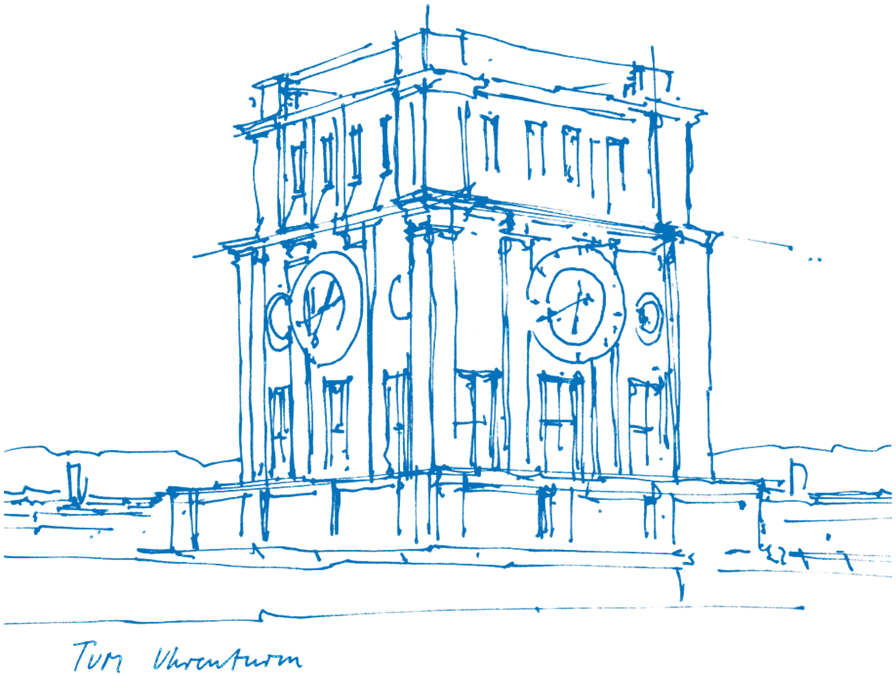
Diskrete Strukturen Tutorium

Jay Zhou
Technische Universität München
Garching b. München, 4. Dezember 2023



Prüfungsanmeldung freigeschaltet!!!

Graphentheorie — Eulertour/Eulerpfad

In einem Eulertour/Eulerpfad wird jede Kante genau einmal besucht.

- Jeder Knoten in einem einfachen zusammenhängenden Graph $G = (V, E)$ mit einem geraden positiven Knotengrad besitzt eine Eulertour.

Graphentheorie — Eulertour/Eulerpfad

Auf Eulertour prüfen:
Alle Knoten haben einen geraden positiven Knotengrad: WENN der Graph ZUSAMMENHÄNGEND ist, dann hat er eine Eulertour.

Sonst hat der Graph keine Eulertour.

Auf Eulerpfad prüfen:
Alle Knoten oder alle außer 2 Knoten haben einen geraden positiven Knotengrad: WENN der Graph ZUSAMMENHÄNGEND ist, dann hat er einen Eulerpfad.

Sonst hat der Graph keine Eulerpfad.

Aufgabe

Aufgabe 7.1 b)

Beweisen Sie, dass jede Graphsequenz $\boxed{(1, \dots, \lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \dots, n-1)}$ realisierbar ist.

Idee: Fallunterscheidung

n Knoten: Sobald $\lfloor \frac{n}{2} \rfloor$ auftaucht, dupliziert man $\lfloor \frac{n}{2} \rfloor$ einmal und dann weiter bis zum $n-1$

$n = 2 \quad (\underset{\lfloor \frac{n}{2} \rfloor}{1} \;\; \underset{n-1}{1})$

$n = 3 \quad (\underset{\lfloor \frac{n}{2} \rfloor}{1} \;\; 1 \;\; \underset{n-1}{2})$

$n = 7 \quad (1 \;\; 2 \;\; \underset{\lfloor \frac{n}{2} \rfloor}{3} \;\; 3 \;\; 4 \;\; 5 \;\; \underset{n-1}{6})$

Basis:

$n = 2 \longrightarrow (1, 1)$ 

$n = 3 \longrightarrow (1, 1, 2)$ 

Schritt: Sei $n \geq 2$ beliebig fixiert

Annahme: $(1, \dots, \lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \dots, n-1)$ realisierbar

Behauptung: $(1, \dots, \lfloor \frac{n+2}{2} \rfloor, \lfloor \frac{n+2}{2} \rfloor, \dots, n+1)$ realisierbar

Beweis: $(1, \dots, \lfloor \frac{n+2}{2} \rfloor, \lfloor \frac{n+2}{2} \rfloor, \dots, \cancel{n+1}) \Rightarrow (0, 1, \dots, \lfloor \frac{n+2}{2} \rfloor - 1, \lfloor \frac{n+2}{2} \rfloor - 1, \dots, n-1)$

Fall 1: n ist gerade

$n := 2k \Rightarrow \lfloor \frac{n+2}{2} \rfloor - 1 = \lfloor \frac{2k}{2} \rfloor = k$

$(0, 1, \dots, k, k, \dots, n-1)$

$(0, 1, \dots, \underbrace{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \dots, n-1)}_{\text{Annahme}}$

$\uparrow$ Ein Knoten

$\square$

Fall 2: n ist ungerade

$n := 2k+1 \Rightarrow \lfloor \frac{n+2}{2} \rfloor - 1 = \lfloor \frac{2k+1}{2} \rfloor = k$

$(0, 1, \dots, k, k, \dots, n-1)$

$(0, 1, \dots, \underbrace{\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \dots, n-1)}_{\text{Annahme}}$

$\uparrow$ Ein Knoten

$\square$

Wir untersuchen ein Wort w, das: - alle Wörter $\in \Sigma^k$ enthält

- Länge $|\Sigma|^k + k - 1$ hat

Aufgabe 7.2

Sei Σ ein endliches Alphabet und $k > 1$ beliebig aber fest. Zeigen Sie: Es gibt ein Wort $w \in \Sigma^*$ der Länge $|w| = |\Sigma|^k + k - 1$, welches jedes Wort der Länge k genau einmal als Faktor enthält, d.h. für jedes $u \in \Sigma^k$ gibt es $x, y \in \Sigma^*$ mit $w = xuy$.

$|2|^3 + 3 - 1 = 10$

Geben Sie speziell für $\Sigma = \{a, b\}$ und $k = 3$ ein solches Wort an.

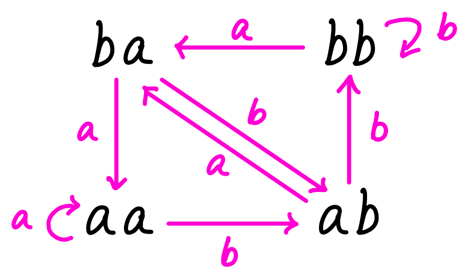
Hinweis: Konstruieren Sie einen geeigneten Graphen mit Knotenmenge $V = \Sigma^{k-1}$.

Fürs Fall

$k = 3$ und $\Sigma = \{a, b\}$

Graph:

Knoten: aa, ab, ba, bb



$\forall v \in V$:

eingehende Kanten: 2

ausgehende Kanten: 2

Zusammenhang: Ja

$\Rightarrow$ Eulertour

$\Rightarrow$ Für jedes Wort der Länge $k-1$ wird das Wort um ein Σ (a oder b) erweitert. Damit ist jedes Wort aus $\Sigma^{k-1}\Sigma = \Sigma^k$ ein mal enthalten.

Beispielwort: aaabbbabaa

aaa	baa
aab	aba
abb	bab
bba	
bbb	

Alle sind einmal enthalten

Länge des Wortes

$2 + 8 = 10 = 2^3 + 3 - 1 = |\Sigma|^k + k - 1$

Beginn (Knoten) — Kantenanzahl (Da jede Kanten ein mal besucht wird)

Allgemein

Man konstruiert einen Graph $G = (V, E)$

$V = \{ \Sigma^{k-1} \}$

$E = \{ a_1 \dots a_{k-1}, a_2 \dots a_k x \mid x \in \Sigma \}$ $\quad aa \xrightarrow{b} \cancel{a}ab$

$\boxed{\text{Es gibt für jeden Knoten genau } k \text{ eingehende und ausgehende Kanten}} + \boxed{\text{Zusammenhang}}$

Eulertour

$\Rightarrow$ Für jedes Wort der Länge $k-1$ wird das Wort um ein Σ erweitert. Damit ist jedes Wort aus $\Sigma^{k-1}\Sigma = \Sigma^k$ ein mal enthalten.

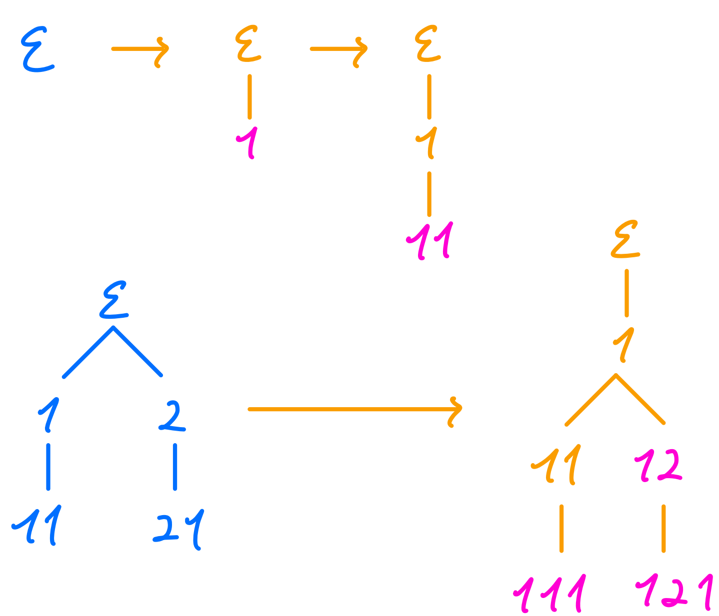
Länge des Wortes: $\underbrace{|\Sigma|^k}_{\substack{\text{Kantenanzahl} \\ \text{(Da jede Kanten ein mal besucht wird)}}} + \underbrace{\underline{k-1}}_{\substack{\text{Beginn} \\ \text{(Knoten)}}}$

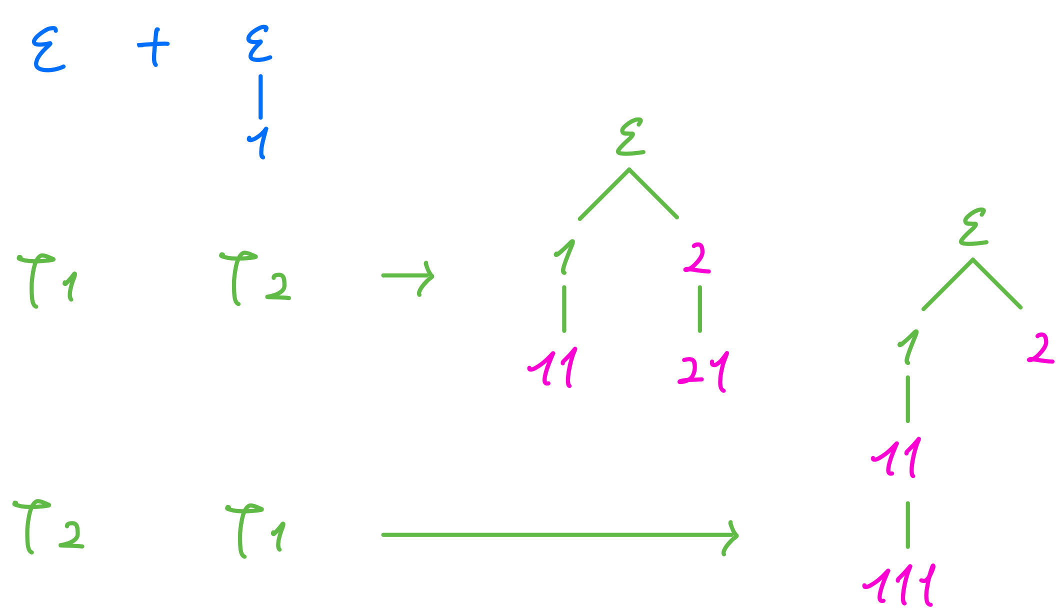
Aufgabe 7.3

Wir betrachten folgende Regeln, um eine Menge von *zulässigen* endlichen Wurzelbäume (V, E) mit $V \subseteq \{1,2\}^*$, $E \subseteq \{(u, u1), (u, u2) \mid u \in \{1,2\}^*\}$ und Wurzel ε zu konstruieren:

- Der Baum $(\{\varepsilon\}, \emptyset)$ ist *zulässig*.
- Ist $T_1 = (V_1, E_1)$ ein *zulässiger* Baum, dann auch T mit $V = \{1u \mid u \in V_1\} \cup \{\varepsilon\}$ und $E = \{(\varepsilon, 1)\} \cup \{(1u, 1v) \mid (u, v) \in E_1\}$.
- Sind $T_1 = (V_1, E_1), T_2 = (V_2, E_2)$ zwei *zulässige* Bäume, dann auch T mit $V = \{11u \mid u \in V_1\} \cup \{2u \mid u \in V_2\} \cup \{\varepsilon, 1\}$ und $E = \{(\varepsilon, 1), (1, 11), (\varepsilon, 2)\} \cup \{(11u, 11v) \mid (u, v) \in E_1\} \cup \{(2u, 2v) \mid (u, v) \in E_2\}$.

Beispiele





Aufgabe 7.3

Wir betrachten folgende Regeln, um eine Menge von *zulässigen* endlichen Wurzelbäume (V, E) mit $V \subseteq \{1,2\}^*$, $E \subseteq \{(u, u1), (u, u2) \mid u \in \{1,2\}^*\}$ und Wurzel ε zu konstruieren:

- Der Baum $(\{\varepsilon\}, \emptyset)$ ist *zulässig*.
- Ist $T_1 = (V_1, E_1)$ ein *zulässiger* Baum, dann auch T mit $V = \{1u \mid u \in V_1\} \cup \{\varepsilon\}$ und $E = \{(\varepsilon, 1)\} \cup \{(1u, 1v) \mid (u, v) \in E_1\}$.
- Sind $T_1 = (V_1, E_1), T_2 = (V_2, E_2)$ zwei *zulässige* Bäume, dann auch T mit $V = \{11u \mid u \in V_1\} \cup \{2u \mid u \in V_2\} \cup \{\varepsilon, 1\}$ und $E = \{(\varepsilon, 1), (1, 11), (\varepsilon, 2)\} \cup \{(11u, 11v) \mid (u, v) \in E_1\} \cup \{(2u, 2v) \mid (u, v) \in E_2\}$.

$$\begin{array}{c} \varepsilon \\ | \\ 1 \end{array}$$

in T_1: ε

$1T_1$: Ein 1 vor jedem Knoten von T_1 hinzufügen

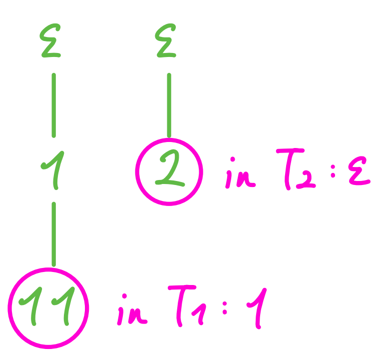
$$\Rightarrow T_{k+1} = \begin{array}{c} \varepsilon \\ | \\ 1 \\ | \\ 1T_k \end{array} = \begin{array}{c} \varepsilon \\ | \\ 1T_k \end{array}$$

Spitze ist 1, denn die Spitze von T_k ist ε

Aufgabe 7.3

Wir betrachten folgende Regeln, um eine Menge von *zulässigen* endlichen Wurzelbäume (V, E) mit $V \subseteq \{1,2\}^*$, $E \subseteq \{(u, u1), (u, u2) \mid u \in \{1,2\}^*\}$ und Wurzel ε zu konstruieren:

- Der Baum $(\{\varepsilon\}, \emptyset)$ ist *zulässig.*
- Ist $T_1 = (V_1, E_1)$ ein *zulässiger* Baum, dann auch T mit $V = \{1u \mid u \in V_1\} \cup \{\varepsilon\}$ und $E = \{(\varepsilon, 1)\} \cup \{(1u, 1v) \mid (u, v) \in E_1\}$.
- Sind $T_1 = (V_1, E_1), T_2 = (V_2, E_2)$ zwei *zulässige* Bäume, dann auch T mit $V = \{11u \mid u \in V_1\} \cup \{2u \mid u \in V_2\} \cup \{\varepsilon, 1\}$ und $E = \{(\varepsilon, 1), (1, 11), (\varepsilon, 2)\} \cup \{(11u, 11v) \mid (u, v) \in E_1\} \cup \{(2u, 2v) \mid (u, v) \in E_2\}$.



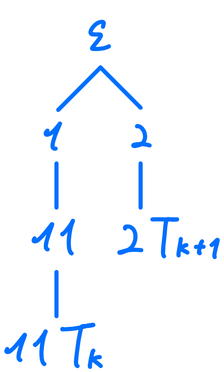
$\downarrow$ $11T_1$:

Ein 11 vor jedem Knoten von T_1 hinzufügen

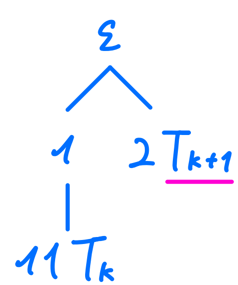
$\downarrow$ $2T_2$:

Ein 2 vor jedem Knoten von T_2 hinzufügen

$\Rightarrow T_{k+2} =$



$=$



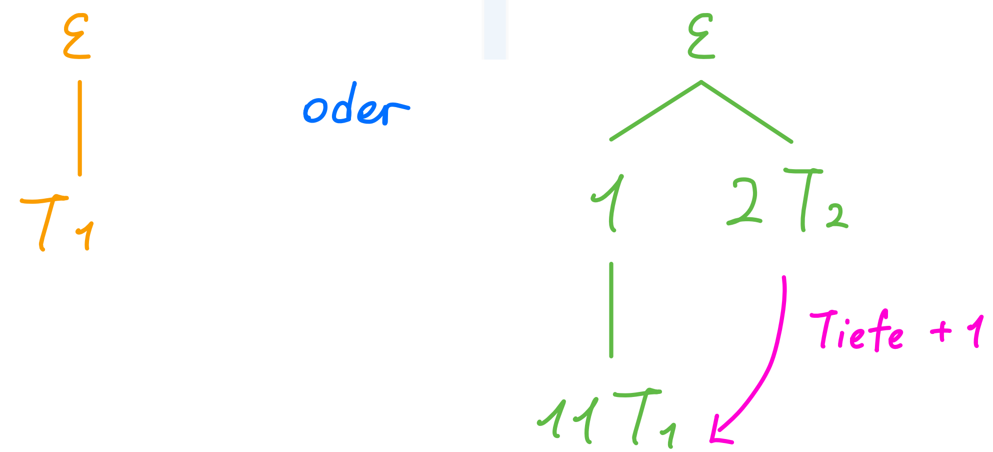
Spitze ist 2
denn die Spitze von T_{k+1} ist ε

Spitze ist 11
denn die Spitze von T_k ist ε

a) kommt danach :)

Sei $N(k)$ die Anzahl der *zulässigen* Bäume, **in denen jeder Pfad von der Wurzel** ε **zu einem Blatt genau die Länge** k **hat**.

(b) Geben Sie eine möglichst einfache Rechenvorschrift an, die $N(k+1)$ mittels der Werte $N(1), \ldots, N(k)$ bestimmt.



Da die Bäume balanciert werden müssen, wäre es optimal, dass der Baum T_k die Tiefe k hat.

Basis: T_0 hat Tiefe 0, T_1 hat Tiefe 1
(ε) (ε – 1)

Schritt: Sei $k \geq 2$ beliebig fixiert

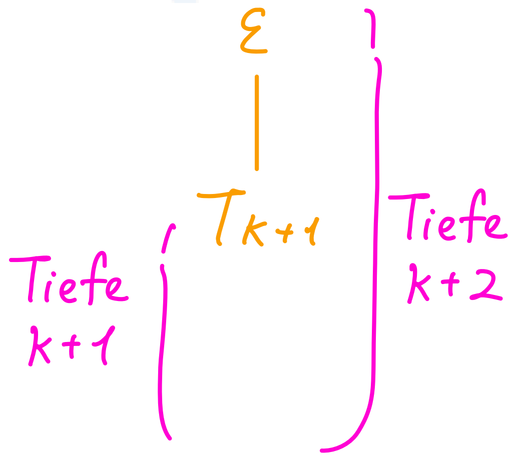
Annahme: T_k hat Tiefe k, T_{k+1} hat Tiefe $k+1$

Behauptung: T_{k+2} hat Tiefe $k+2$

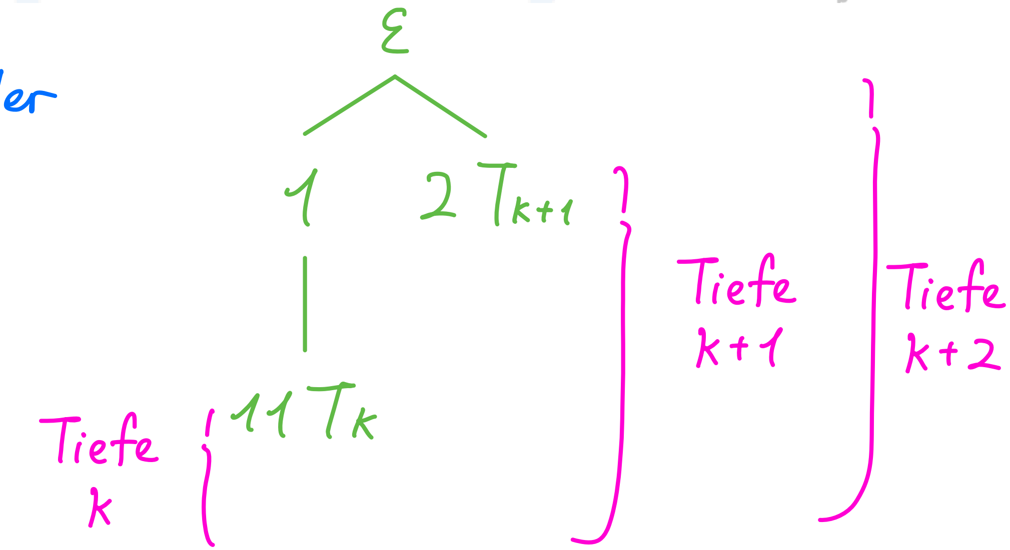
Sei $N(k)$ die Anzahl der *zulässigen* Bäume, **in denen jeder Pfad von der Wurzel ε zu einem Blatt genau die Länge k hat**.

(b) Geben Sie eine möglichst einfache Rechenvorschrift an, die $N(k+1)$ mittels der Werte $N(1), \ldots, N(k)$ bestimmt.

T_{k+2} :



oder



Anzahl $= N(k+1)$

Anzahl $= N(k) \cdot N(k+1)$

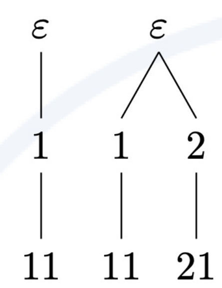
$$N(k+2) = N(k+1) + N(k) \cdot N(k+1)$$

Sei $N(k)$ die Anzahl der *zulässigen* Bäume, **in denen jeder Pfad von der Wurzel ε zu einem Blatt genau die Länge k hat.**

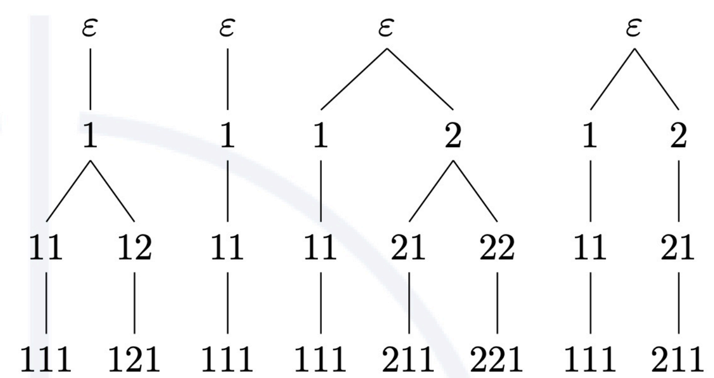
(a) Bestimmen Sie $N(k)$ für $k = 0, 1, 2, 3$ explizit, indem Sie die entsprechenden zulässigen Bäume zeichnen.

Base Cases
$$\begin{cases} N(0) = 1 & \varepsilon \\ N(1) = 1 & \varepsilon - 1 \end{cases}$$

$$N(2) = N(1) + N(0) \cdot N(1) = 1 + 1 \cdot 1 = 2$$



$$N(3) = N(2) + N(1) \cdot N(2) = 2 + 1 \cdot 2 = 4$$



Sei $N(k)$ die Anzahl der *zulässigen* Bäume, **in denen jeder Pfad von der Wurzel ε zu einem Blatt genau die Länge k hat.**

(c) Zeigen Sie mittels geeigneter Induktion, dass $N(k) \geq 2^{(k-4)^2}$ für alle $k \geq 4$ gilt.

Gegeben: $N(k+2) = N(k+1) + N(k) \cdot N(k+1)$

$\Rightarrow$ Zweistufig

Basis: $k = 4, 5$ $\quad N(4) = N(3) + N(2) \cdot N(3) = 12$

$$\vee$$

$$2^{(4-4)^2} = 1 \quad \checkmark$$

$$N(5) = N(4) + N(3) \cdot N(4) = 60$$

$$\vee$$

$$2^{(5-4)^2} = 2 \quad \checkmark$$

Schritt: Sei $k \geq 6$ beliebig fixiert

Annahme: $N(k) \geq 2^{(k-4)^2}$ und $N(k+1) \geq 2^{(k-3)^2}$

Behauptung: $N(k+2) \geq 2^{(k-2)^2}$

Beweis :

$$N(k+2) = N(k+1) + N(k)\cdot N(k+1)$$

$$= N(k+1)\cdot(N(k)+1)$$

$a > b > c$ $\quad \geq N(k+1)\cdot N(k)$

$$\geq 2^{(k-3)^2}\cdot 2^{(k-4)^2}$$

$$= 2^{(k-3)^2+(k-4)^2}$$

$$= 2^{k^2-6k+9+k^2-8k+16}$$

$$= 2^{2k^2-14k+25}$$

$$f(k) := 2k^2-14k+25$$

$$N(k+2) \geq 2^{(k-2)^2} = 2^{k^2-4k+4}$$

$$g(k) := k^2-4k+4$$

$\Rightarrow$ Zu Zeigen : $\forall k \geq 6.\ f(k) \geq g(k)$

$f'(k) = 4k - 14$ $\qquad g'(k) = 2k - 4$

$f''(k) = 4$ — ab Minimum $k = \frac{7}{2}$ ist die Funktion monoton wachsend

$g''(k) = 2$ — ab Minimum $k = 2$ ist die Funktion monoton wachsend

$f(k)$ wächst schneller als $g(k)$

und an der Stelle $k = 6$ gilt

$$f(6) = 13 < 16 = g(6) \ ⚡$$

und an der Stelle $k = 7$ gilt

$$f(7) = 25 = 25 = g(7) \ ✓$$

$\Rightarrow \forall k \geq 7.\ f(k) \geq g(k)$, bewiesen für $\forall k \geq 7$

$\square$

Falls $k = 6$, (explizit beweisen)

$$N(6) = N(5) + N(4)\cdot N(5) = 780$$

✓

$$2^{(6-4)^2} = 16$$

bewiesen für $k = 6$

Bonusaufgabe

(a) Sei $G = (V, E)$ ein endlicher Baum mit genau $n \geq 2$ Knoten und aufsteigend sortierter Gradsequenz $(d_1, d_2, \ldots, d_n)$.

Zeigen Sie:

(i) $\sum_{i=1}^{n} d_i = 2n - 2$.

(ii) $d_1 = d_2 = 1$.

(i)

① Induktion

Basis: für $n=1$ gilt $\sum_{i=1}^{1} d_i = 2 \cdot 1 - 2 = 0$

Baum: $\bullet$

Schritt Annahme: $\sum_{i=1}^{n} d_i = 2 \cdot n - 2$

Behauptung: $\sum_{i=1}^{n+1} d_i = 2 \cdot n$

Beweis: $\sum_{i=1}^{n+1} d_i = \sum_{i=1}^{n} d_i + d_{n+1}$

$= 2 \cdot n - 2 + 1 + 1$

1 Knoten hinzufügen
Es darf nur 1 Kante mit einem anderen Knoten verbunden werden.

neu hinzugefügte Knoten

Kanten zu dem anderen Knoten, Grad +1

$= 2 \cdot n$

$\square$

② Theorie

Baum: $|E| = |V| - 1$

Handschlaglemma: $2|E| = \sum_{i=1}^{n} d_i$

$2(|V|-1) = \sum_{i=1}^{n} d_i \Rightarrow 2n - 2 = \sum_{i=1}^{n} d_i$

$\square$

(ii)

VL-Skript Abschnitt 143

Jeder Baum hat mind. 2 Blätter
mit $|V| \geq 2$

Blatt: Grad ≤ 1

Zusammenhang:

$\nexists v \in V.\ deg(v) = 0$

$\Rightarrow$ Wegen Sortierung: $d_1 = d_2 = 1$

$\square$

Bonusaufgabe

(b) Zeigen Sie: jede aufsteigend sortierte Gradsequenz $(d_1, d_2, \ldots, d_n)$ mit $n \geq 2$, $d_1 = d_2 = 1$ und $\sum_{i=1}^{n} d_i = 2n - 2$ lässt sich durch einen Baum realisieren.

Zu Zeigen

Hinweis: Verwenden Sie Induktion nach $n \geq 2$. Nehmen Sie für den Induktionsschritt an, dass, sollte die gegebene Folge durch einen Baum realisierbar sein, es dann auch stets einen Baum mit dieser Gradfolge gibt, in dem ein Knoten vom maximalen Grad stets ein Blatt als Nachbarn hat.

Basis: $n = 2$ •—• $(1,1)$

① $d_1 = d_2 = 1$ ✓
② $\sum_{i=1}^{2} d_i = 2 \cdot 2 - 2 = 2$ ✓
③ d_{max} verbunden mit d_{min} ✓

Schritt

Annahme: Jede aufst. sortierte Gradsequenz mit den 3 Eigenschaften lässt sich durch einen Baum realisieren.

① $d_1 = d_2 = 1$
② $\sum_{i=1}^{n} d_i = 2 \cdot n - 2$
③ d_{max} verbunden mit d_{min}

Behauptung: Jede aufst. sortierte Gradsequenz mit den 3 Eigenschaften lässt sich durch einen Baum realisieren.

① $d_1 = d_2 = 1$
② $\sum_{i=1}^{n+1} d_i = 2 \cdot n$
③ d_{max} verbunden mit d_{min}

Beweis :

Wir fügen einen Knoten hinzu.

Da $\sum_{i=1}^{n+1} d_i - \sum_{i=1}^{n} d_i = 2$, ist die $\sum_{v \in V} deg(v)$ um 2 erhöht.

Da $d_{min} = 1$, wird also eine Kante zwischen dem neuen Knoten und einem anderen hinzugefügt.

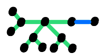
Fall 1 : neu •——• d_{max}

① $d_1 = d_2 = 1$ ✓ die bleiben noch da
② $\sum_{i=1}^{n+1} d_i = 2 \cdot n$ ✓ $2n-2+2 = 2n$
③ d_{max} verbunden mit d_{min} ✓ das neue Knoten ist d_{min}

$\square$

Fall 2 : neu •——• d_{min}

① $d_1 = d_2 = 1$ ✓ d_2 bleibt, $d_{min} \rightarrow d_1$
② $\sum_{i=1}^{n+1} d_i = 2 \cdot n$ ✓ $2n-2+2 = 2n$
③ d_{max} verbunden mit d_{min} ✗ Gegenbeispiel



Fall 3 : neu •——• weder d_{min} noch d_{max}

① $d_1 = d_2 = 1$ ✓ die bleiben noch da
② $\sum_{i=1}^{n+1} d_i = 2 \cdot n$ ✓ $2n-2+2 = 2n$
③ d_{max} verbunden mit d_{min} ✓ die Kante bleibt da

$\square$

EINEN... realisierbar!

$\square$

Fragen?