Technische Universität Braunschweig

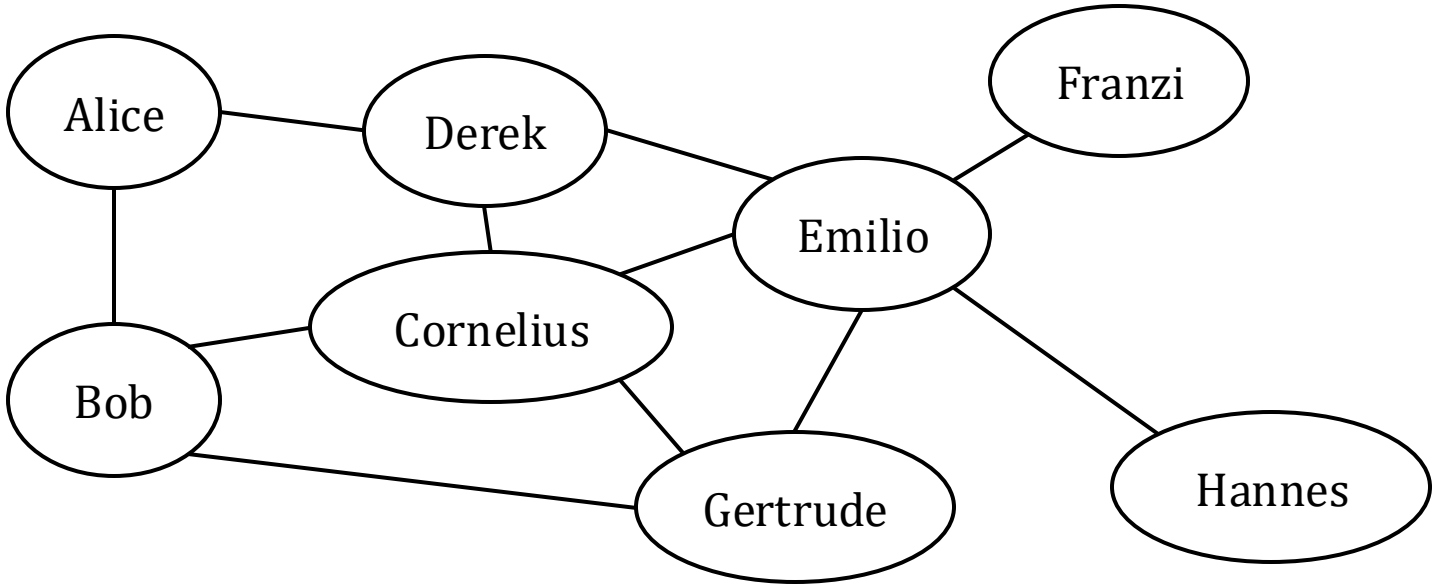
# Parametrisierte Algorithmen

Arne Schmidt

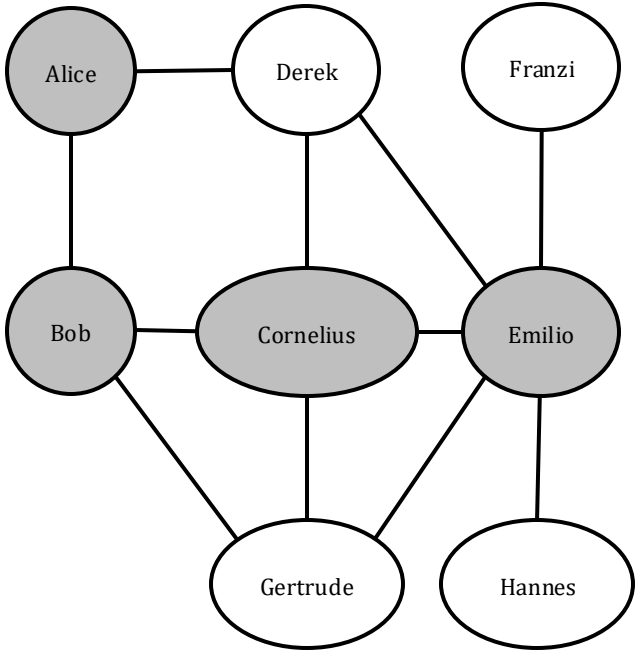
# Recap – Bar Fight Problem

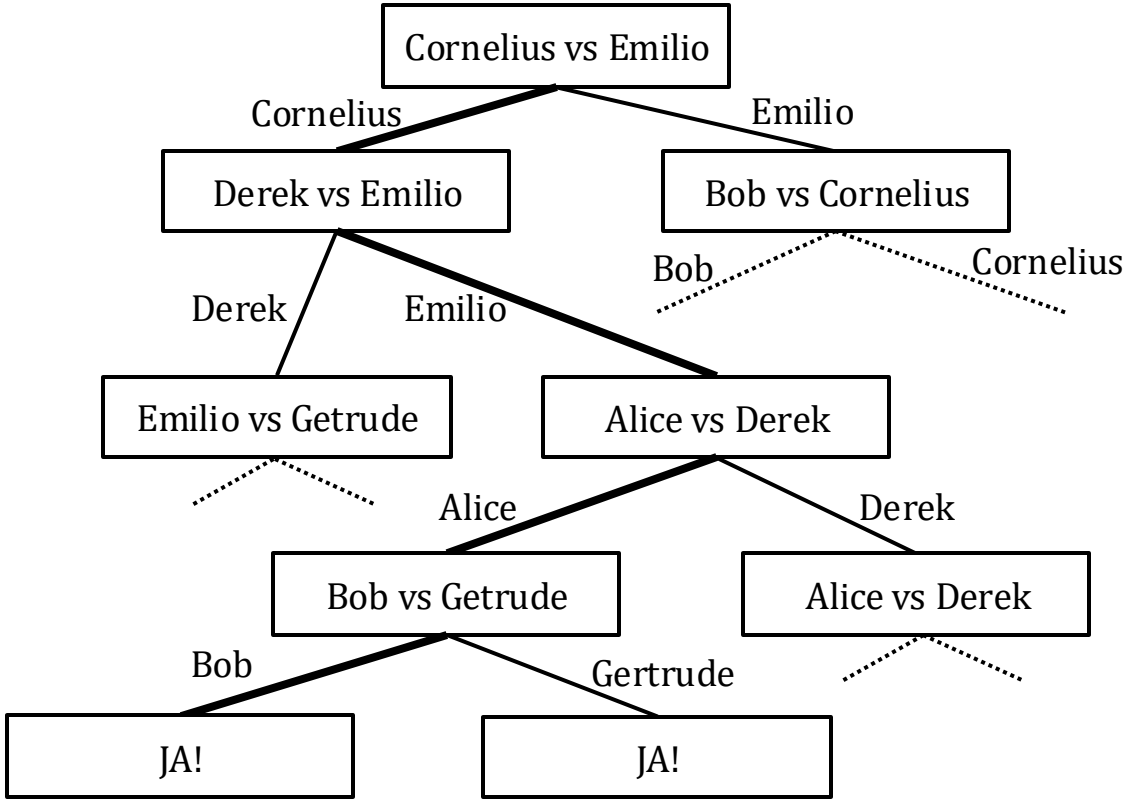
# Ansatz 2: Bounded Tree Search

**Lemma 1.3:**
Das Bar-Fight-Problem lässt sich in Zeit $O\left(2^k \cdot k \cdot n\right)$ lösen.



# Ansatz 2: Bounded Tree Search



# Ansatz 2: Bounded Tree Search

**Lemma 1.3:**
Das Bar-Fight-Problem lässt sich in Zeit $O\left(2^k \cdot k \cdot n\right)$ lösen.

**Idee:**

- Betrachte eine beliebige Kante $\{u, v\} \in E$. Einer der beiden Endknoten **muss** im Vertex Cover liegen.
- Angenommen, $v$ ist im Vertex Cover, dann können wir $v$ samt Kanten entfernen und testen, ob der restliche Graph eine Lösung mit $k' = k - 1$ besitzt.
- Angenommen, $u$ ist im Vertex Cover, dann können wir $u$ samt Kanten entfernen und testen, ob der restliche Graph eine Lösung mit $k' = k - 1$ besitzt.

Rekursionstiefe ist maximal $k$, Verzweigungsgrad der Rekursion ist 2.
→ $O\left(2^k\right)$ Knoten im Suchbaum.
Den jeweils neuen Graphen zu berechnen dauert maximal $O(n + m) = O(kn)$ Zeit.

# Kapitel 3 – Bounded Tree Search

# Allgemeine Idee

Für eine gegebene Instanz $(I, k)$, triff $O\big(f(k)\big)$ viele Entscheidungen, unter den Bedingungen:

- $f: \mathbb{N} \to \mathbb{N}$ ist eine berechenbare Funktion.
- $k$ für jede Instanz der Entscheidung kleiner wird
- $I$ eine ja-Instanz ist, gdw. unter allen Entscheidungen eine ja-Instanz ist.
- Die Entscheidungen in poly. Zeit im Input berechnet werden können.

Damit ergibt sich eine Laufzeit von:

$$O\big(\nu(k) \cdot \text{poly}(|I|, k)\big)$$

Dabei ist $\nu(k)$ die Anzahl besuchter Knoten im Entscheidungsbaum.

# Allgemeine Idee

Damit ergibt sich eine Laufzeit von:

$$O\big(\nu(k) \cdot \text{poly}(|I|, k)\big)$$

Dabei ist $\nu(k)$ die Anzahl besuchter Knoten im Entscheidungsbaum.

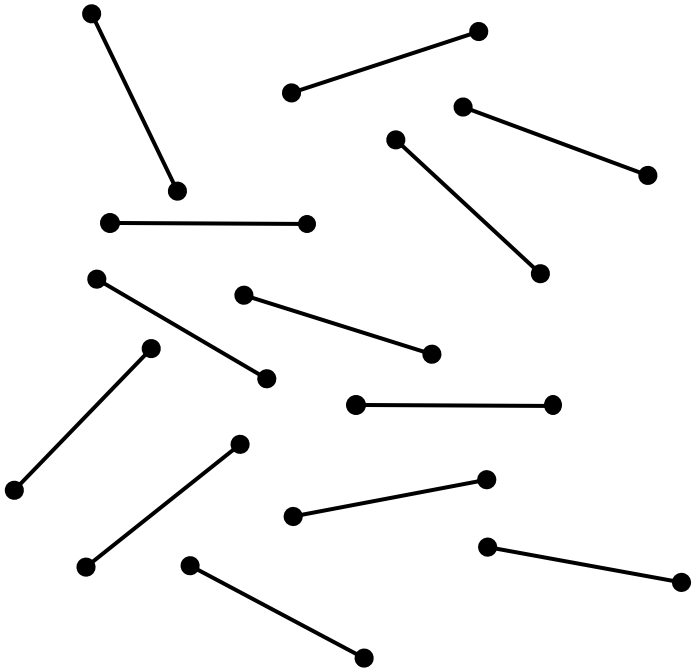
Für Vertex Cover ist mit vorheriger Strategie $\nu(k) = 2^k$

Also eine Laufzeit von $O\big(2^k kn\big)$

Geht das schneller?!

# Kapitel 3.1 – Vertex Cover

# Unabhängige Kanten



Alle Kanten sind unabhängig.

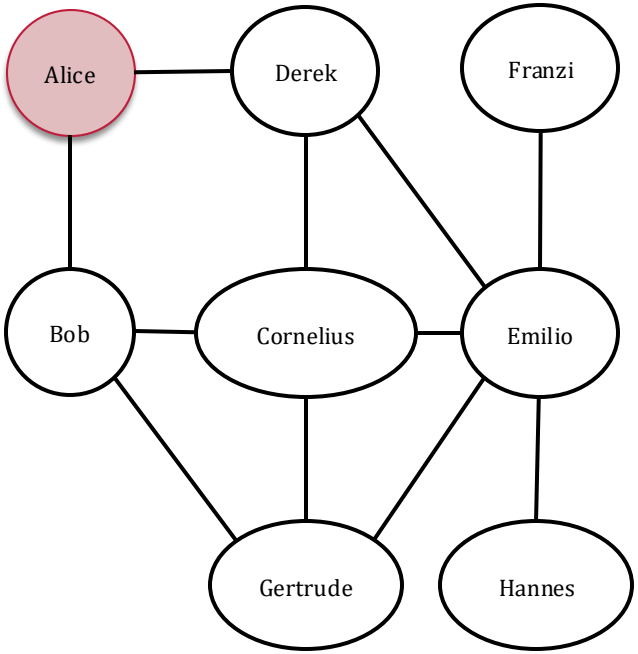
Wie schnell kann man Vertex Cover hier lösen?
→ Linear Zeit!
Instanz ist genau dann eine Ja-Instanz, wenn $k \geq |E|$

Ist jeder Knotengrad 1, lässt sich VC in Polyzeit lösen!

Auf schweren Instanzen muss es also Knoten mit Grad mind. 2.

# Knoten von Grad 2

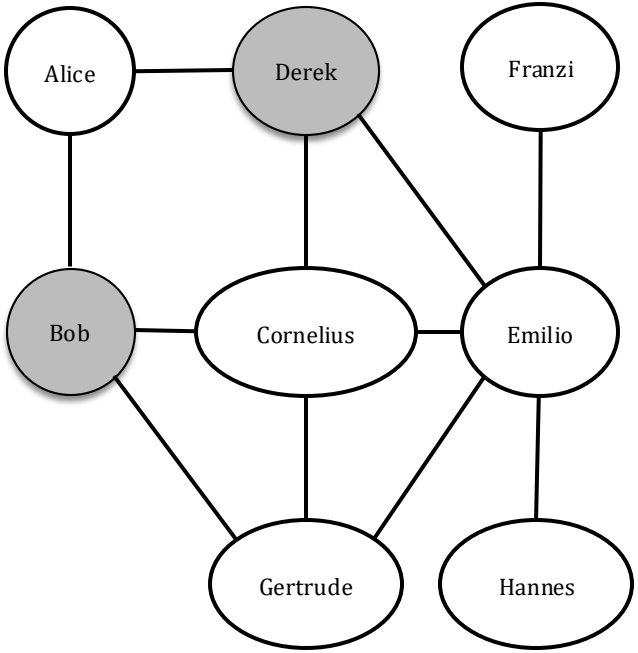
$k = 4$

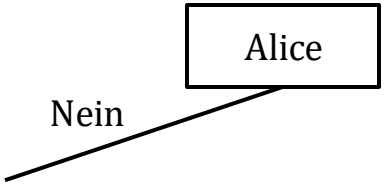



Alice

# Knoten von Grad 2

$k = 4$



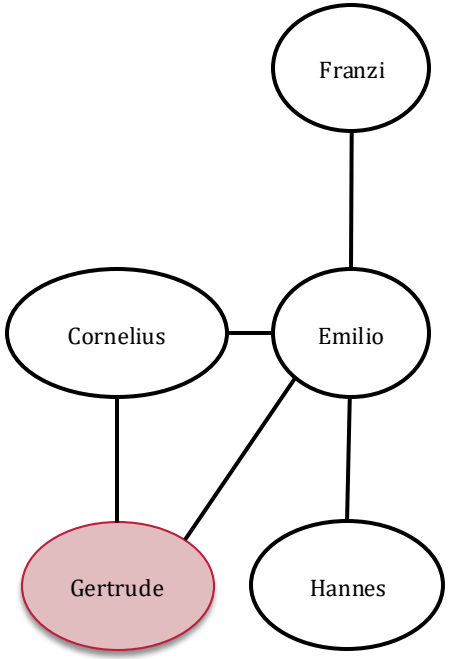



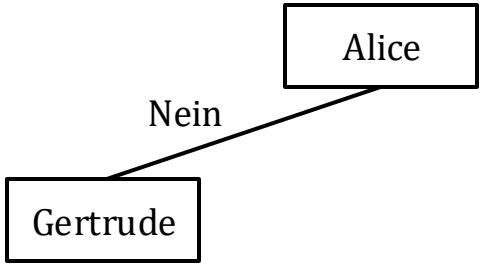

Ist Alice nicht im Vertex Cover, müssen ihre Nachbarn es sein!

Der Parameter reduziert sich um den Grad des Knotens!

# Knoten von Grad 2

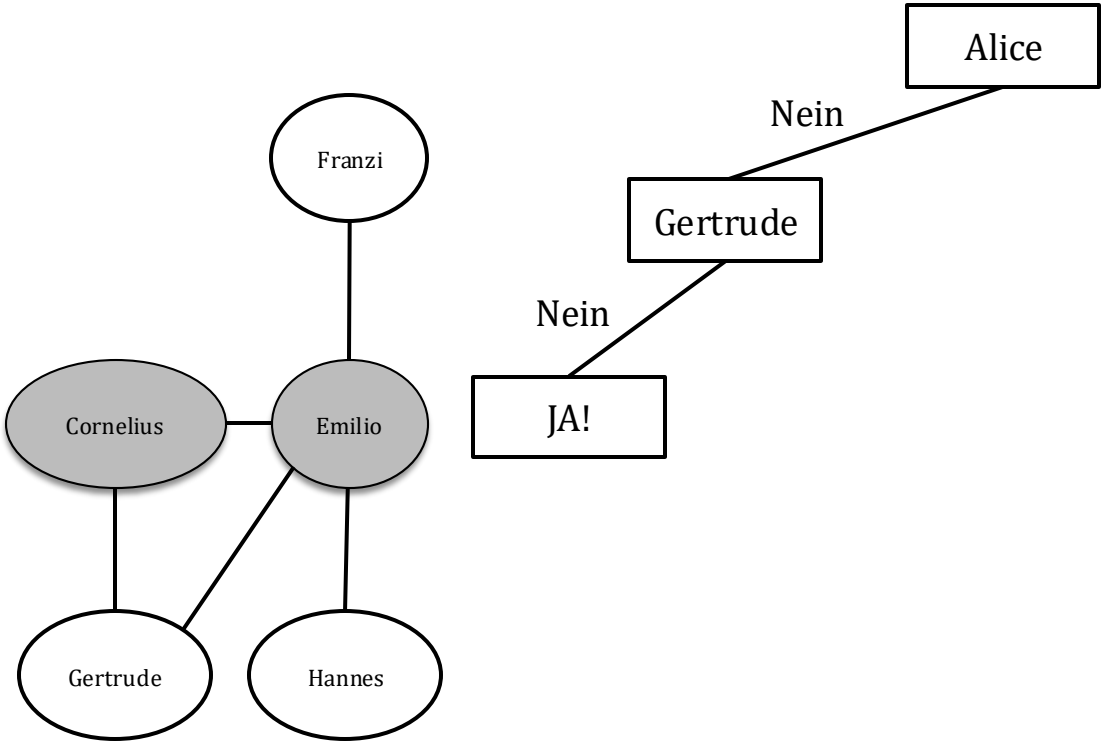
$k = 2$



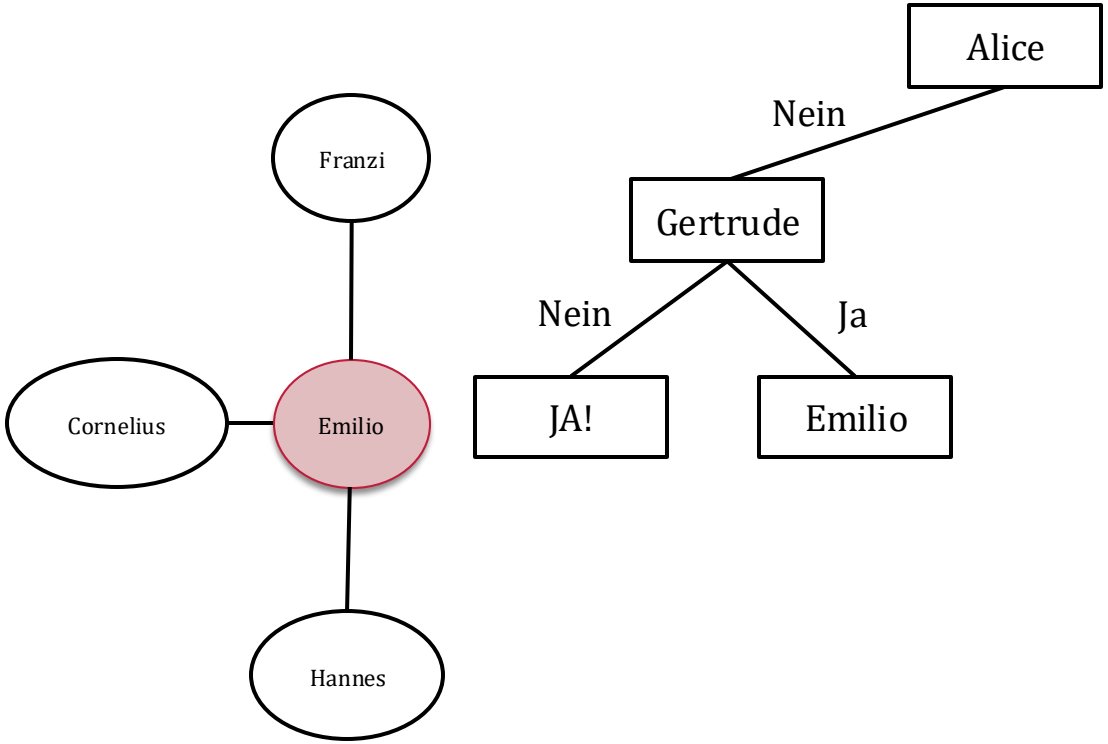
# Knoten von Grad 2

$k = 2$



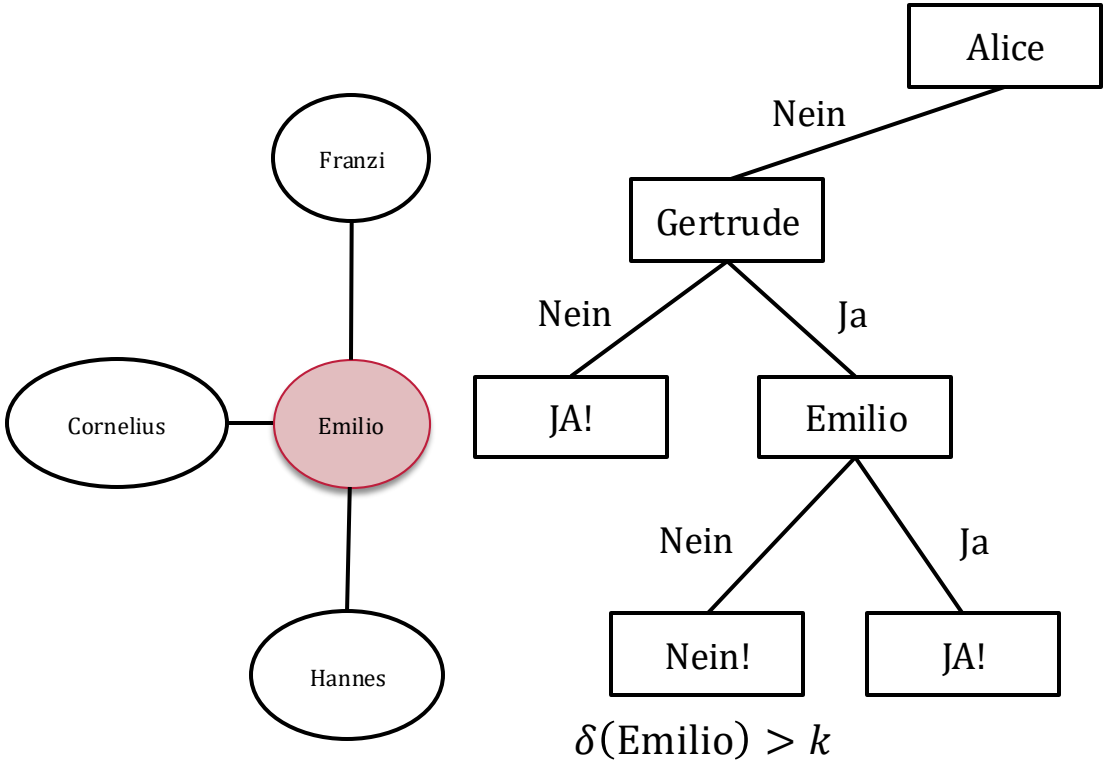
# Knoten von Grad 2

$k = 1$



# Knoten von Grad 2

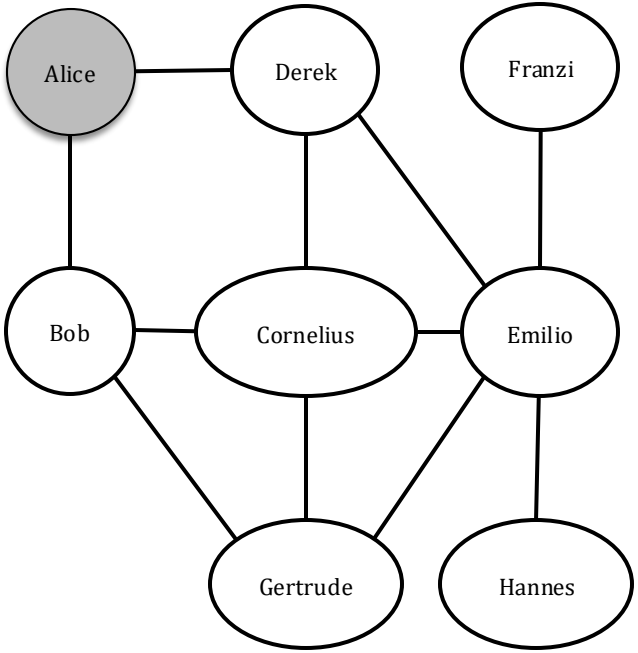
$k = 1$

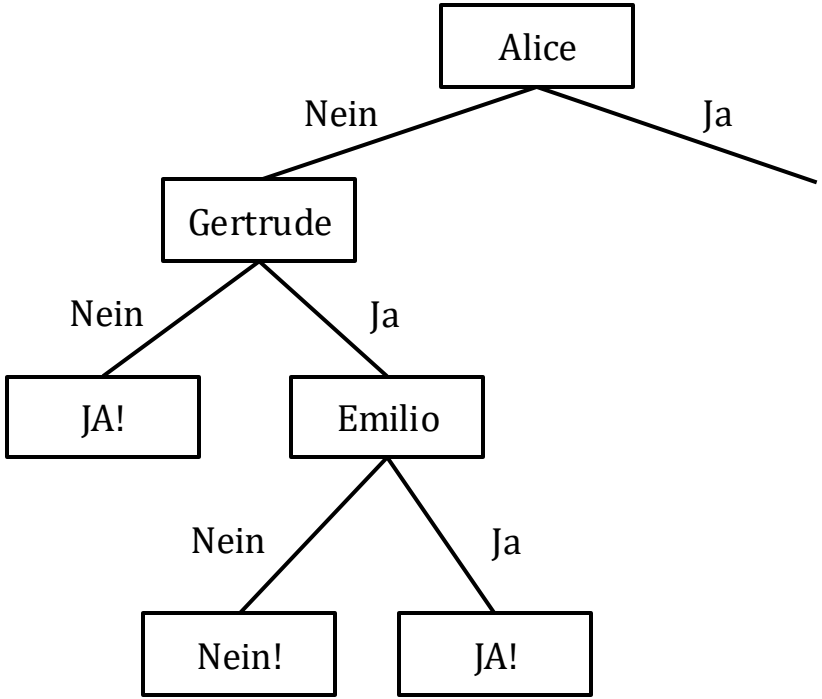



$\delta(\text{Emilio}) > k$

# Knoten von Grad 2

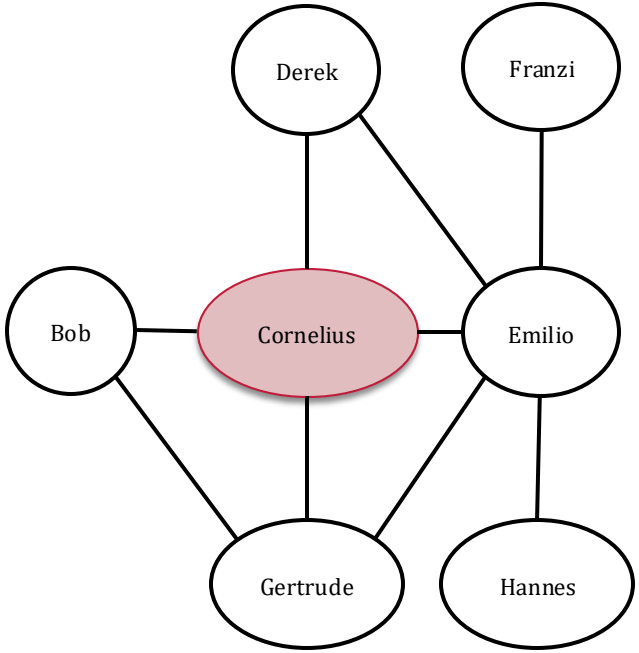
$k = 4$

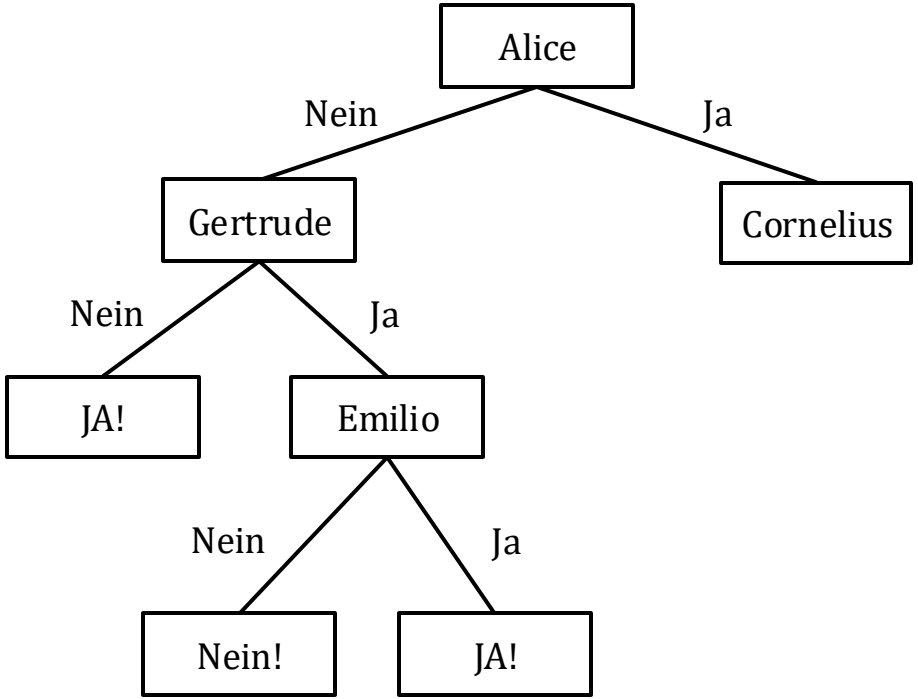


# Knoten von Grad 2

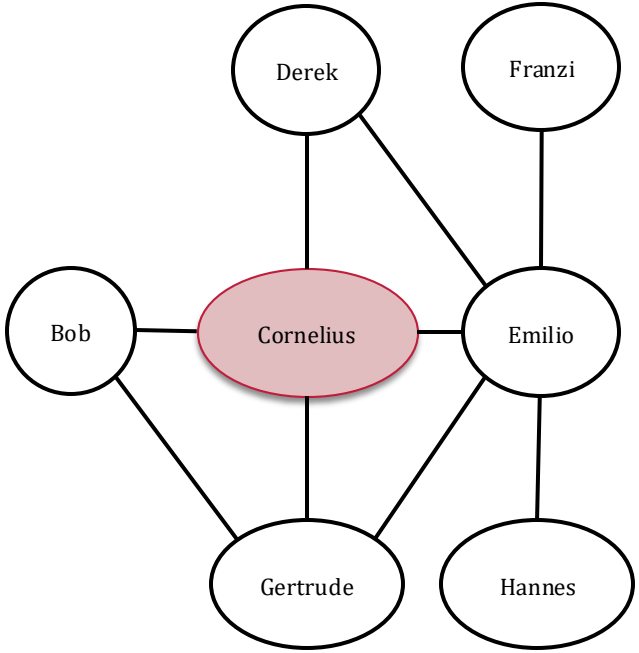
$k = 3$

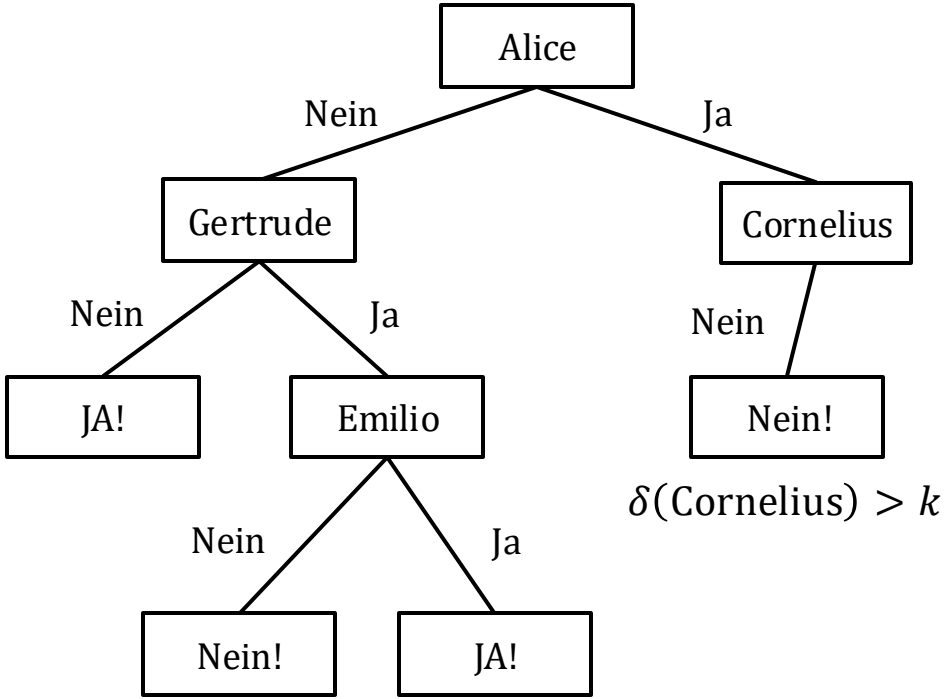


# Knoten von Grad 2

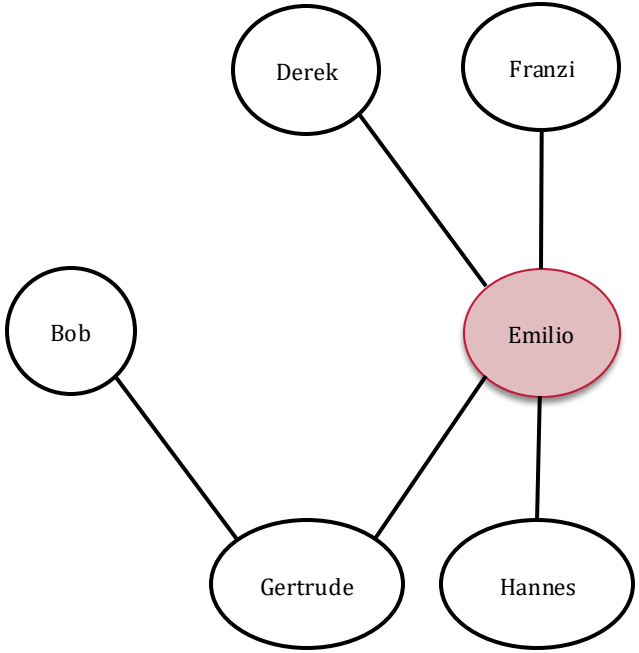
$k = 3$

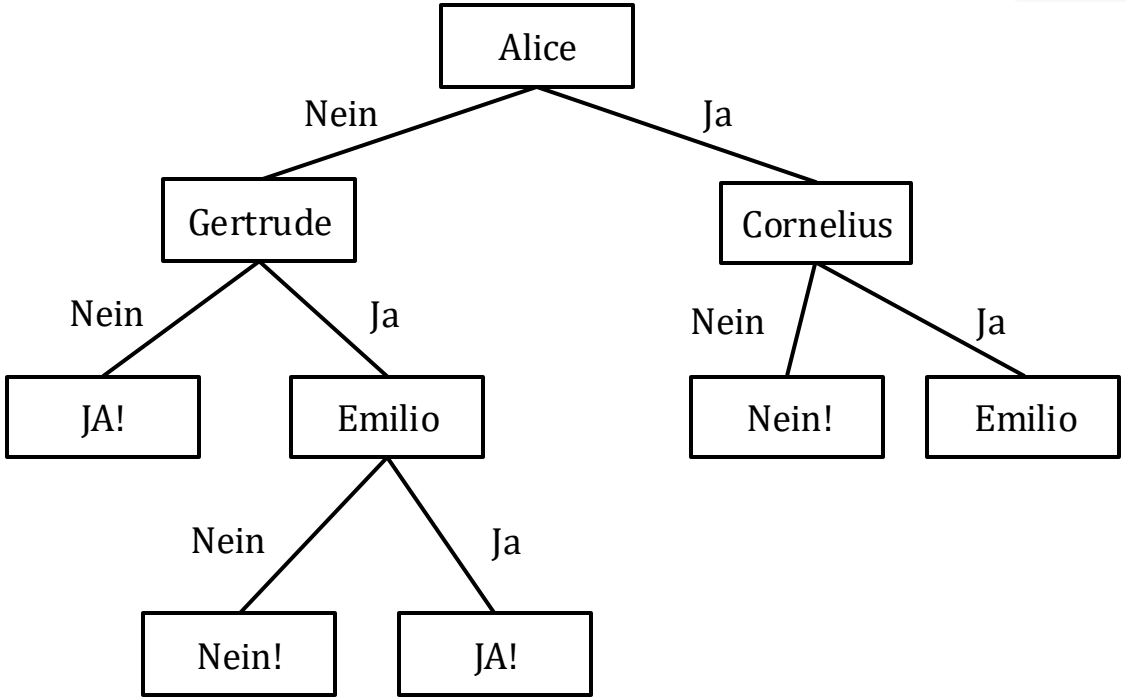


# Knoten von Grad 2

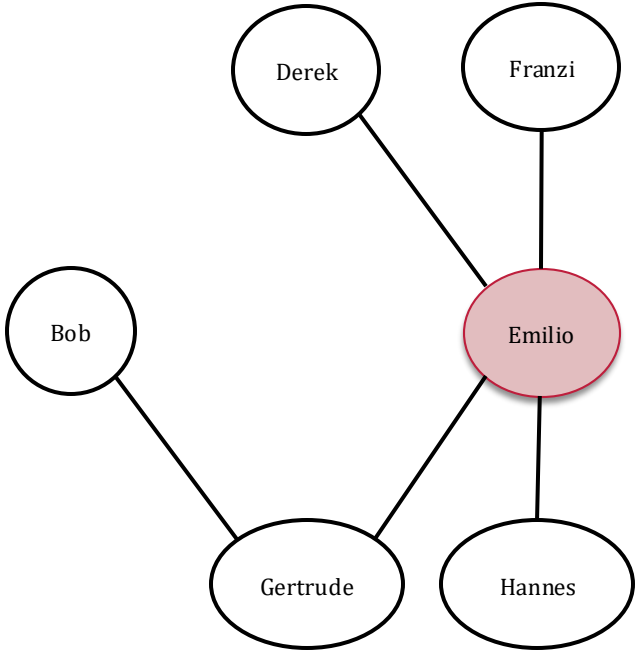
$k = 2$

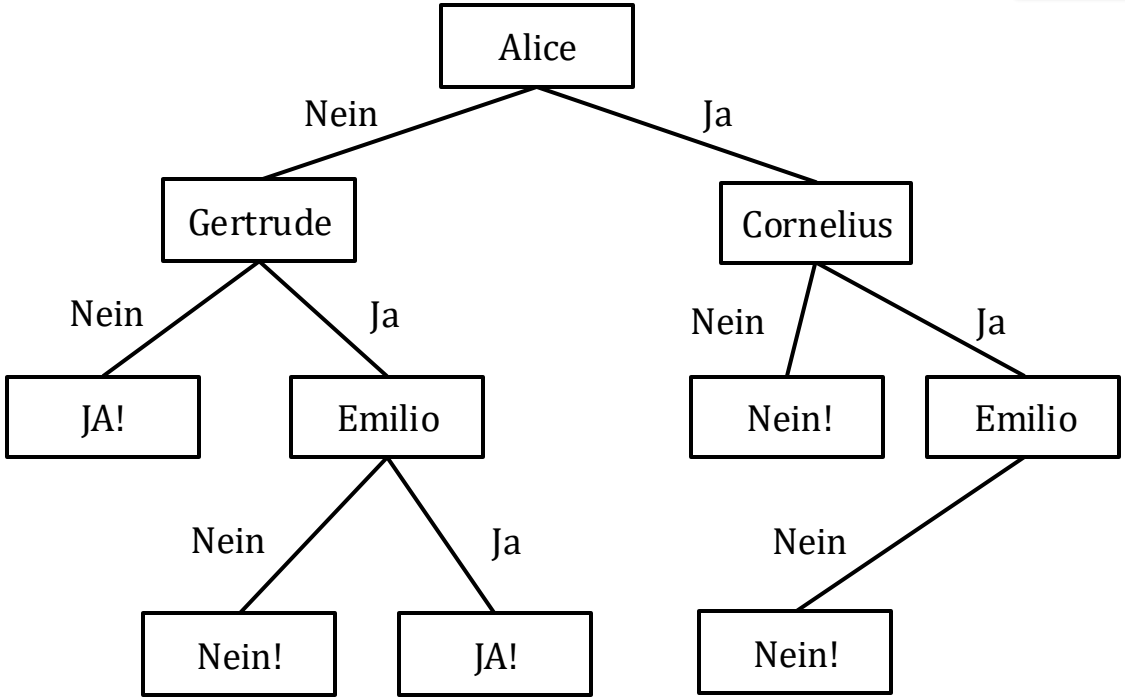


# Knoten von Grad 2

$k = 2$

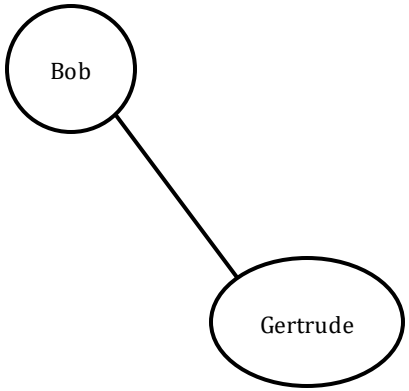


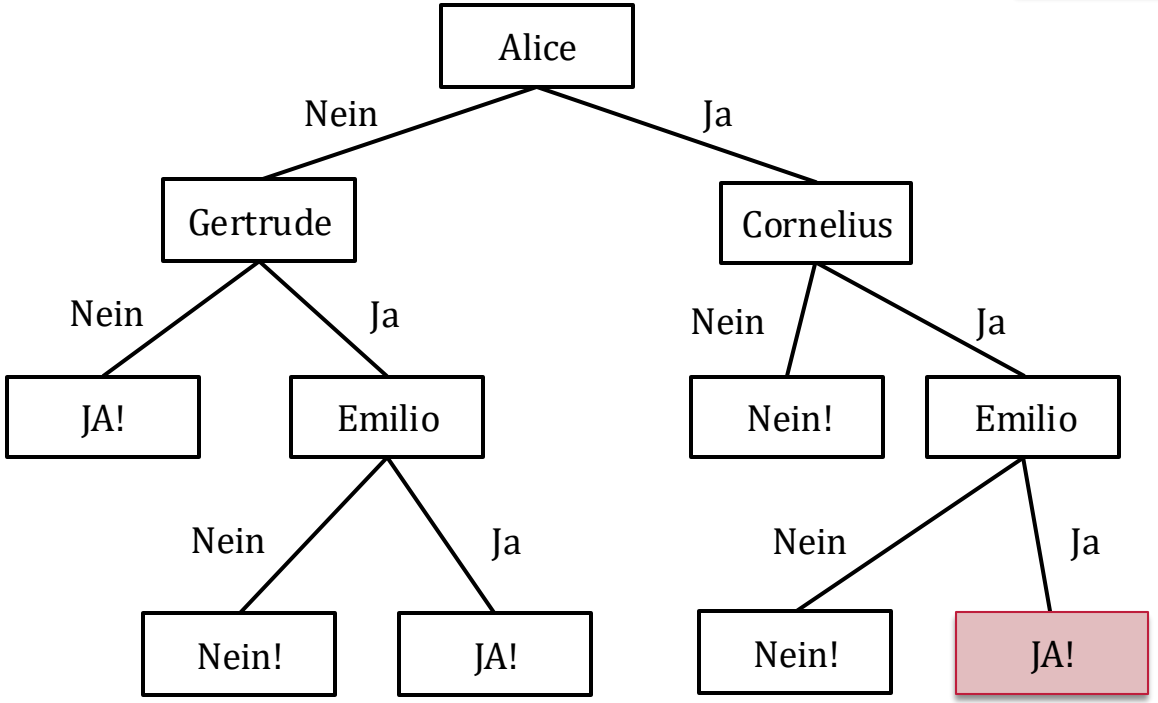




$\delta(\text{Emilio}) > k$

# Knoten von Grad 2

$k = 1$

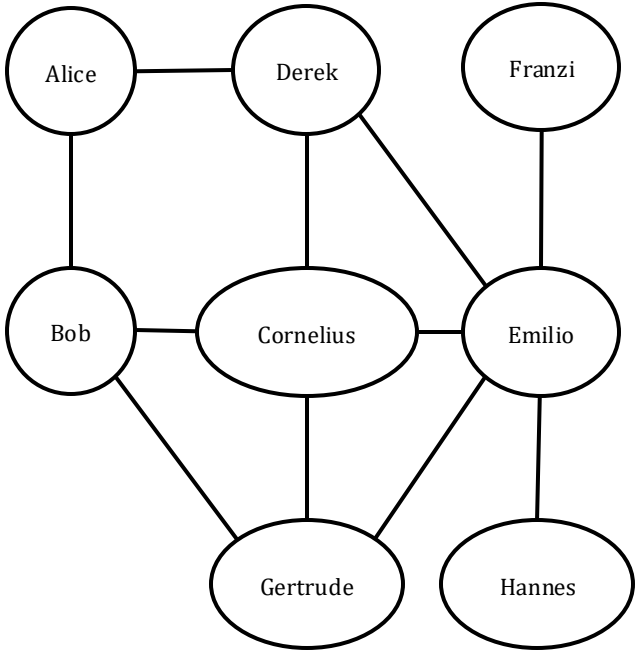


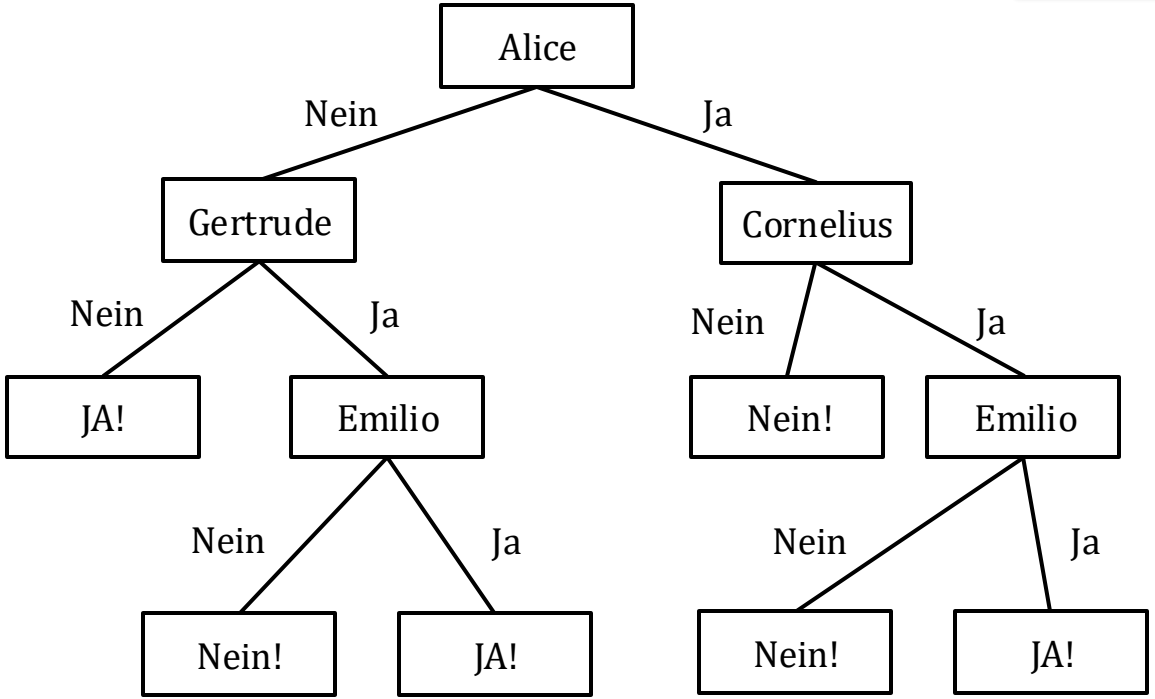




Restgraph nur Knoten von Grad 1

# Knoten von Grad 2

$k = 4$



# Laufzeitanalyse

Sei $T(k)$ die Anzahl an betrachteten Instanzen im Entscheidungsbaum für Parameter $k$.
Wir betrachten zwei Fälle für einen Knoten $v \in V$:

1. Knoten $v$ ist im Vertex Cover: Lösche $v$ samt Kanten und fahre mit $k' = k - 1$ fort.
2. Knoten $v$ ist *nicht* im Vertex Cover: Lösche Nachbarknoten samt inzidenter Kanten und fahre mit $k' = k - \delta(v) \leq k - 2$ fort.

**Damit gilt im Worst-Case:**

$$T(k) = T(k-1) + T(k-2)$$

**Magic Inspiration:**
Such kleinstes $\lambda \in \mathbb{R}^+$, sodass $T(k) \in O(\lambda^k)$.

**Damit, löse:**

$$\lambda^k \geq \lambda^{k-1} + \lambda^{k-2} \text{ bzw. } \lambda^2 \geq \lambda^1 + 1 \text{ bzw. } \boldsymbol{\lambda^2 - \lambda^1 - 1 \geq 0}$$

# Lösen quadratischer Gleichungen

$$\lambda^2 - \lambda^1 - 1 \geq 0$$

$$\Leftrightarrow \lambda^2 - \lambda^1 + \frac{1}{4} - \frac{1}{4} - 1 \geq 0$$

$$\Leftrightarrow \left(\lambda - \frac{1}{2}\right)^2 - \frac{5}{4} \geq 0$$

$$\Leftrightarrow \lambda \geq \frac{1}{2} \pm \frac{\sqrt{5}}{2}$$

$$\overset{\lambda>0}{\Longrightarrow} \lambda \geq \frac{1}{2} + \frac{\sqrt{5}}{2} \approx 1.6181$$

Das Vertex Cover Problem lässt sich in Zeit $O\left(1.6181^k kn\right)$ Zeit lösen.

# Noch mehr Zeit sparen?

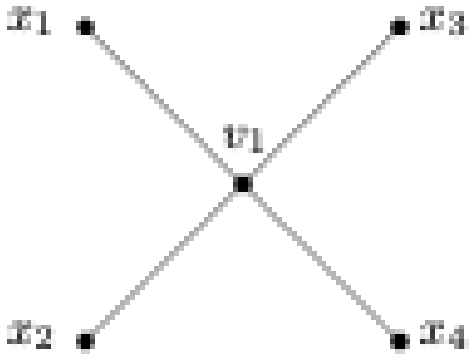
Können wir garantieren, dass wir immer Knoten mit hohem Grad wählen können?

Dafür müssten wir zeigen, dass Graphen mit nur niedrigen Knoten einfach zu lösen sind!

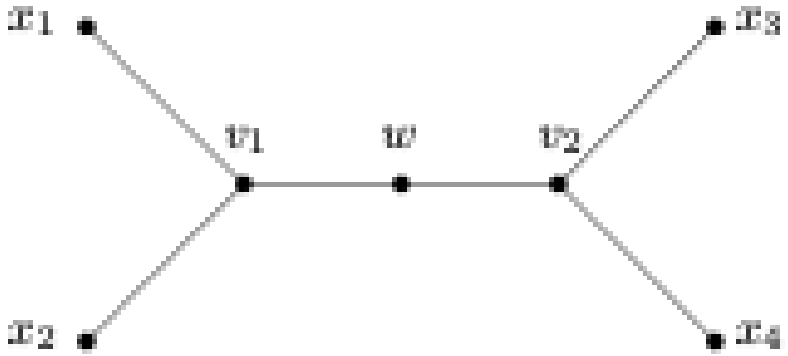
# Maximaler Knotengrad 3

Jede Instanz $(G, k)$ von VC mit Maximalgrad $\Delta > 3$ kann in eine Instanz $(G', k')$ von VC mit Maximalgrad $\Delta = 3$ transformiert werden, sodass

$(G, k)$ ist ja-Instanz gdw. $(G', k')$ ist ja-Instanz




$(G, k)$




$(G', k+1)$

Vertex Cover auf Graphen mit maximalem Grad 3 ist NP-vollständig.

# Maximaler Knotengrad 2

**Lemma 3.1:**
Sei $G$ ein Graph mit Maximalgrad 2. Das Vertex Cover Problem lässt sich in polynomieller Zeit auf solchen Graphen lösen.

Beweis: Selbst / Übung

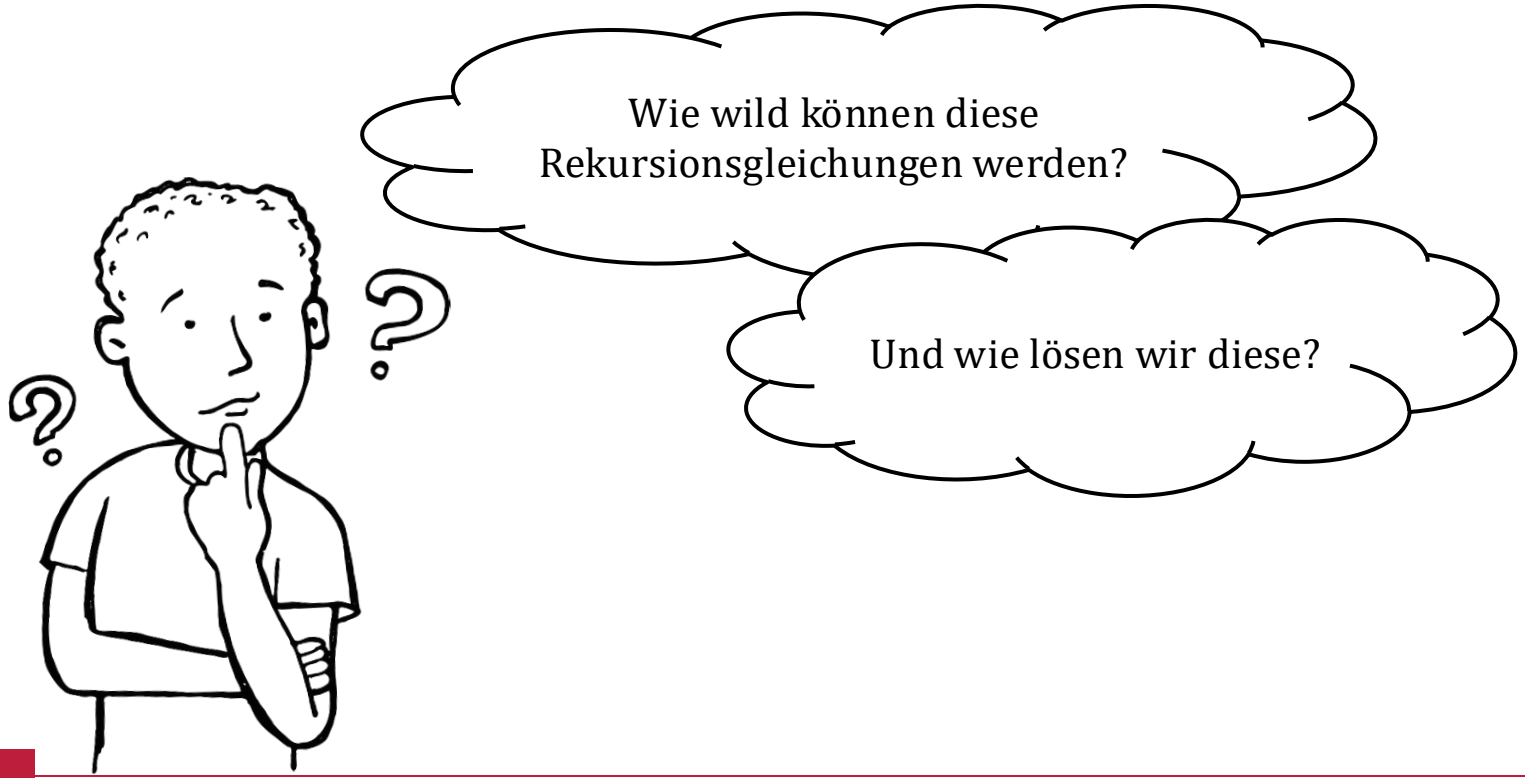
Damit können wir immer einen Knoten von Grad 3 in unserem Algorithmus wählen.
Dadurch erhalten wir

$$T(k) = T(k-1) + T(k-3)$$

Wir haben also maximal $O\big(T(k)\big) = O\big(1.4656^k\big)$ Knoten im Entscheidungsbaum.

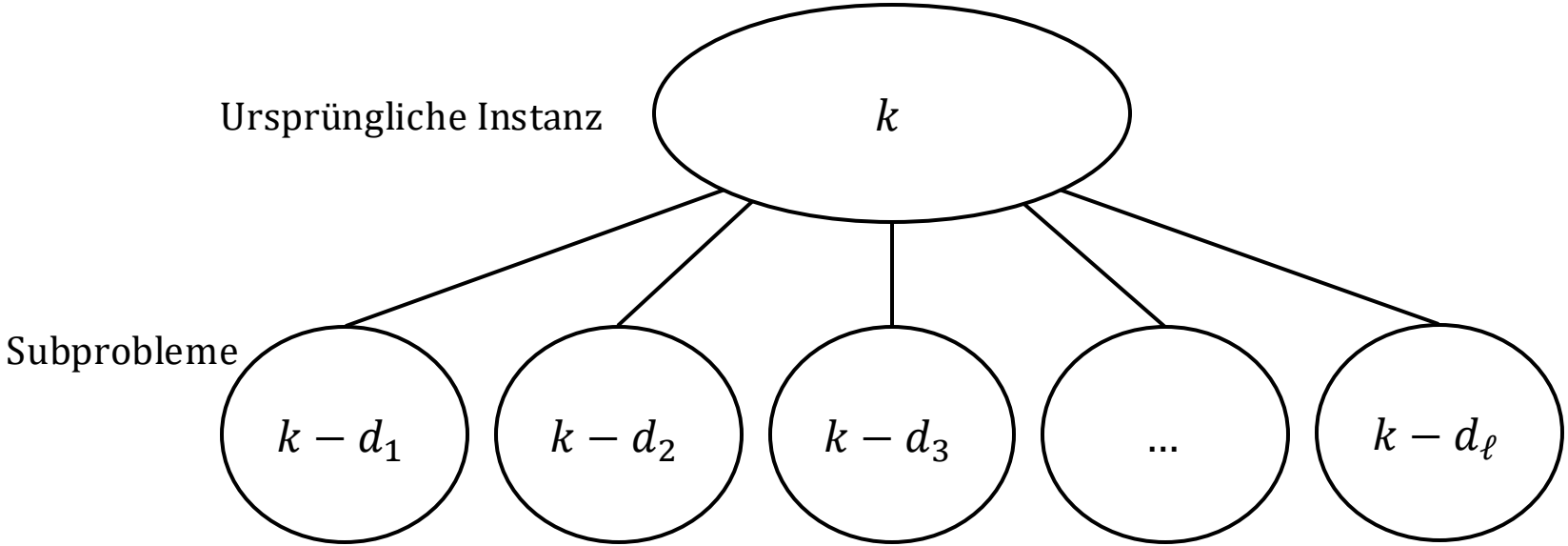
**Theorem 3.2:**
Vertex Cover lässt sich in Zeit $O\big(1.4656^k kn\big)$ Zeit lösen.

# Rekursionsgleichungen lösen



# Kapitel 3.2 – Lösen von Rekursionsgleichungen

# Branching Algorithmen




**Definition 3.3:**
Wird der branching Algorithmus rekursiv $\ell \geq 2$ Mal mit Parametern $k - d_i, i \in \{1, \dots, \ell\}$ aufgerufen, so heißt $(d_1, \dots, d_\ell)$ der **Branching-Vektor** der Rekursion

# Laufzeiten

**Lemma 3.4**
Sei $(d_1, d_2, \dots, d_\ell)$ ein Branching-Vektor eines rekursiven Branching-Algorithmus $\mathcal{A}$. Dann ist die Laufzeit von $\mathcal{A}$ beschränkt durch $T(k) \in O\left(\lambda_0^k\right)$, wobei $\lambda_0$ die Nullstelle des **charakteristischen Polynoms**

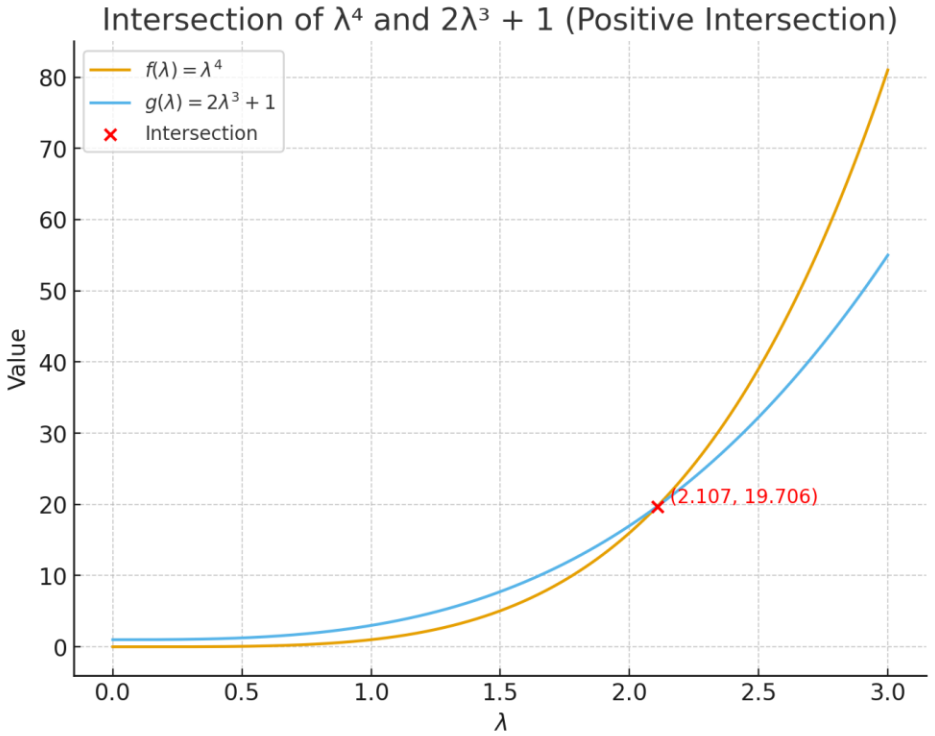
$$P(\lambda) = \lambda^d - \lambda^{d-d_1} - \dots - \lambda^{d-d_\ell}$$

mit $d \coloneqq \max(d_1, d_2, \dots, d_\ell)$ ist.

**Lemma 3.5**
Ein charakteristisches Polynom $P(\lambda) = \lambda^d - \lambda^{d-d_1} - \dots - \lambda^{d-d_\ell}$ mit $d \coloneqq \max(d_1, d_2, \dots, d_\ell)$ besitzt genau eine positive (reelle) Nullstelle $\lambda_0$.

# Nullstellen

**Lemma 3.5**
Ein charakteristisches Polynom $P(\lambda) = \lambda^d - \lambda^{d-d_1} - \cdots - \lambda^{d-d_\ell}$ mit $d \coloneqq \max(d_1, d_2, \ldots, d_\ell)$ besitzt genau eine positive (reelle) Nullstelle $\lambda_0$.



# Beispiele für Nullstellen

| $(i,j)$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | 2.0000 | 1.6181 | 1.4656 | 1.3803 | 1.3248 | 1.2852 |
| 2 | | 1.4143 | 1.3248 | 1.2721 | 1.2366 | 1.2107 |
| 3 | | | 1.2560 | 1.2208 | 1.1939 | 1.1740 |
| 4 | | | | 1.1893 | 1.1674 | 1.1510 |
| 5 | | | | | 1.1487 | 1.1348 |
| 6 | | | | | | 1.1225 |

$\lambda_0$ für den Branching-Vektor $(i,j)$.
(Aufgerundet)

# Beispiele für einfache Rekursionsgleichungen

$T(k) = T(k-1) + T(k-1)$

$T(k) = T(k-2) + T(k-2)$

$T(k) = 2\,T(k-i), i \geq 1$
- Bestimme $\lambda^i - 2 = 0$
- $\Leftrightarrow \lambda = 2^{1/i}$

$\rightarrow T(k) \in O\left(2^{k/i}\right)$

$T(k) = T(k-1) + T(k-2)$

$T(k) = T(k-2) + T(k-4)$

$T(k) = T(k-i) + T(k-2i), i \geq 1$
- Bestimme $\lambda^{2i} - \lambda^i - 1 = 0$
- Substituiere $\phi = \lambda^i$
- Bestimme $\phi^2 - \phi - 1 = 0$
- Berechne $\lambda = \phi^{1/i}$

$\rightarrow T(k) \in O\left(\lambda^k\right)$

# Quadtratische Funktion – Nullstellen

$ax^2 + bx + c = 0$:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

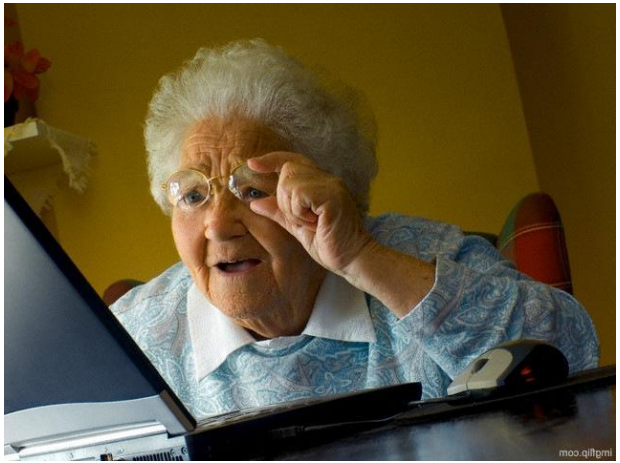
# Kubische Funktion – Nullstellen

$ax^3 + bx^2 + cx + d = 0$:

$$x = \sqrt[3]{\left(\frac{bc}{6a^2} - \frac{d}{2a} - \frac{b^3}{27a^3}\right) + \sqrt{\left(\frac{bc}{6a^2} - \frac{d}{2a} - \frac{b^3}{27a^3}\right)^2 + \left(\frac{c}{3a} - \frac{b^2}{9a^2}\right)^3}} + \sqrt[3]{\left(\frac{bc}{6a^2} - \frac{d}{2a} - \frac{b^3}{27a^3}\right) - \sqrt{\left(\frac{bc}{6a^2} - \frac{d}{2a} - \frac{b^3}{27a^3}\right)^2 + \left(\frac{c}{3a} - \frac{b^2}{9a^2}\right)^3}} - \frac{b}{3a}$$

# Quartische Funktion – Nullstellen

$ax^4 + bx^3 + cx^2 + dx + e = 0$:



$$x_1 = -\frac{b}{4a} + \frac{1}{2}\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12ad\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}} - \frac{1}{2}\sqrt{\frac{b^2}{2a^2} - \frac{4b}{3a} - \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} - \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54} - \frac{-a^3 + 4abd - 8c}{4\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}}}}$$

$$x_2 = -\frac{b}{4a} + \frac{1}{2}\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}} + \frac{1}{2}\sqrt{\frac{b^2}{2a^2} - \frac{4b}{3a} - \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} - \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54} + \frac{-a^3 + 4abd - 8c}{4\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}}}}$$

$$x_3 = -\frac{b}{4a} + \frac{1}{2}\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}} - \frac{1}{2}\sqrt{\frac{b^2}{2a^2} - \frac{4b}{3a} - \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} - \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54} + \frac{-a^3 + 4abd - 8c}{4\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}}}}$$

$$x_4 = -\frac{b}{4a} + \frac{1}{2}\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}} + \frac{1}{2}\sqrt{\frac{b^2}{2a^2} - \frac{4b}{3a} - \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} - \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54} + \frac{-a^3 + 4abd - 8c}{4\sqrt{\frac{b^2}{4a^2} - \frac{2c}{3a} + \frac{\sqrt[3]{2}\left(b^2 - 3ac + 12d\right)}{3a\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}} + \frac{\left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd + \sqrt{-4\left(b^2 - 3ac + 12d\right)^3 + \left(2b^3 - 9abc + 27c^2 + 27a^2d - 72bd\right)^2}\right)^{\frac{1}{3}}}{54}}}}$$

# Grad 5

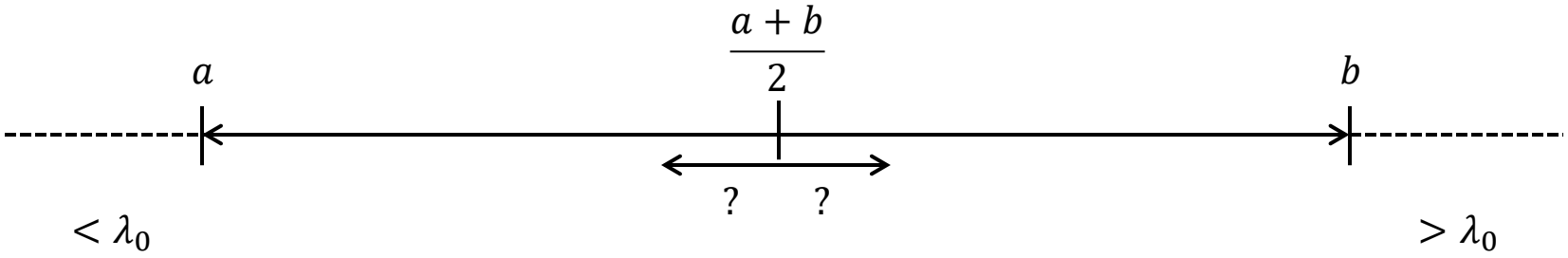
Es existiert keine geschlossene Form für Polynome von Grad $\geq 5$.

Wie finden wir dann unser $\lambda_0$?

Numerisch!

- Newton-Verfahren
- Binäre Suche

# Binäre Suche




Für ein Intervall $[a, b]$ teste, ob $\lambda_0 > \frac{a+b}{2}$ oder $\lambda_0 \leq \frac{a+b}{2}$.
Such dann im entsprechenden Intervall weiter.

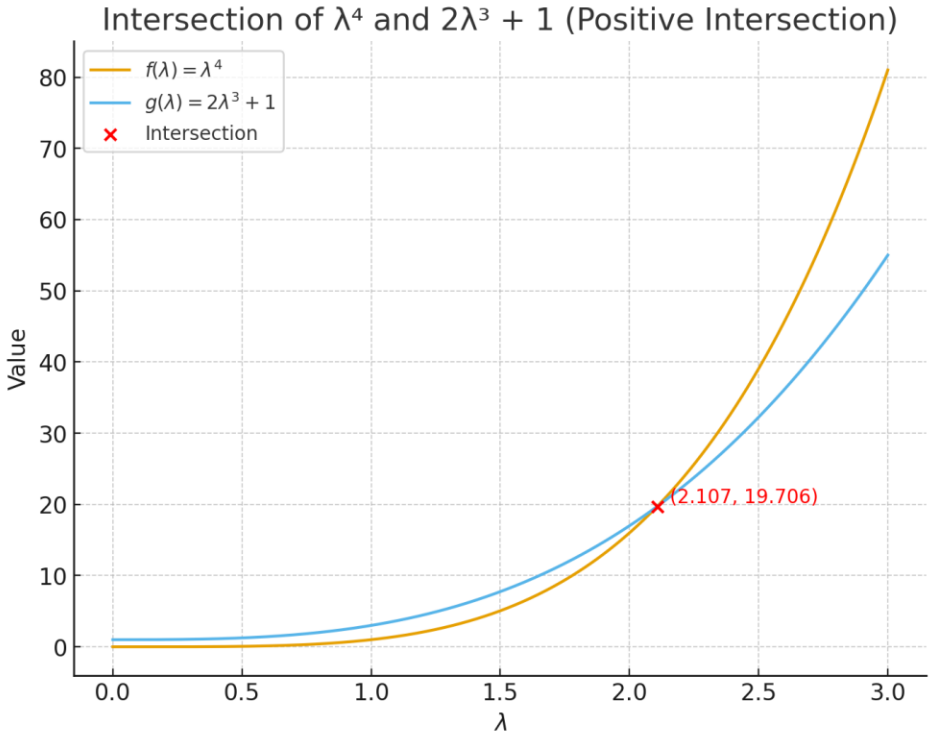
Ist $|b - a| < \varepsilon$ für ein festes $\varepsilon > 0$, gib $b$ zurück.

**Invariante**: $\lambda_0 \in [a, b]$
Mit welchem $a, b$ starten wir?

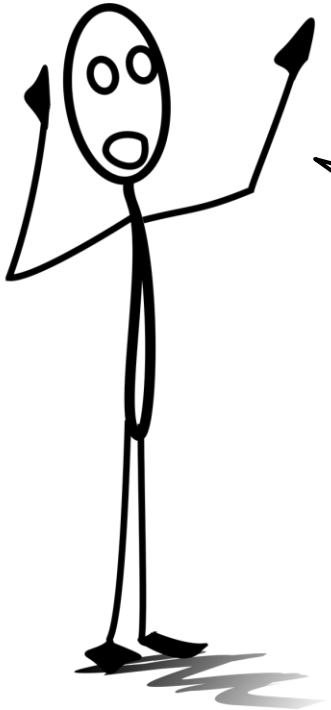
# Schranken für $\lambda_0$

**Lemma 3.5**
Ein charakteristisches Polynom $P(\lambda) = \lambda^d - \lambda^{d-d_1} - \dots - \lambda^{d-d_\ell}$ mit $d \coloneqq \max(d_1, d_2, \dots, d_\ell)$ besitzt genau eine positive (reelle) Nullstelle $\lambda_0$.

**Lemma 3.6**
Sei $P(\lambda) = \lambda^d - \sum_{i=1}^{d} a_i \lambda^{d-i}$ mit $d \in \mathbb{N}$, $\forall i \in \{1, 2, \dots, d\}$: $a_i \in \mathbb{N}$. Dann ist die Nullstelle $\lambda_0 \in [1, \max a_i + 1]$



# Bemerkung



Vereinfachte Annahme: Der Algorithmus brancht immer gleich oder kann direkt lösen.

Ggf. Ergibt es Sinn mehrere Branching-Strategien zu mischen und bei jeder Instanz noch Kernelization zu nutzen!

<u>Beispiel Vertex Cover:</u>
Wir haben nur mit Grad 3 abgeschätzt (worst case). Teilweise können aber Knoten mit höherem Knoten zum Branchen genutzt werden!

# Nächste Woche: Feedback Vertex Set

# Feedback Vertex Set

Lösche Knoten, damit der entstehende Graph kreisfrei wird.

Wie kann man hier Bounded Tree Search nutzen?