Algorithmen und Wahrscheinlichkeit

Woche 4

Ilya Maier

Minitest

Minitest

Password:

Färbungen

Färbung eines Graphen (V, E) **mit** k **Farben**: eine Abbildung $c : V \to [k]$ s.d. $c(u) \neq c(v)$ für alle Kanten $\{u, v\} \in E$

bzw. $V = V_1 \,\dot{\cup}\, \dots \,\dot{\cup}\, V_k$, wobei V_i keine Kanten enthält, V_i := Farbklasse

Chromatische Zahl $\chi(G)$: minimale Anzahl Farben, die für eine Färbung von G benötigt wird.

$$\chi(G) \leq k \iff G \text{ ist } k\text{-partit}$$

Gegeben ein Graph $G = (V, E)$**, gilt** $\chi(G) \leq k$?

$\underline{k = 2}$: In $O(|V| + |E|)$ Zeit mit BFS (keine ungeraden Kreise)

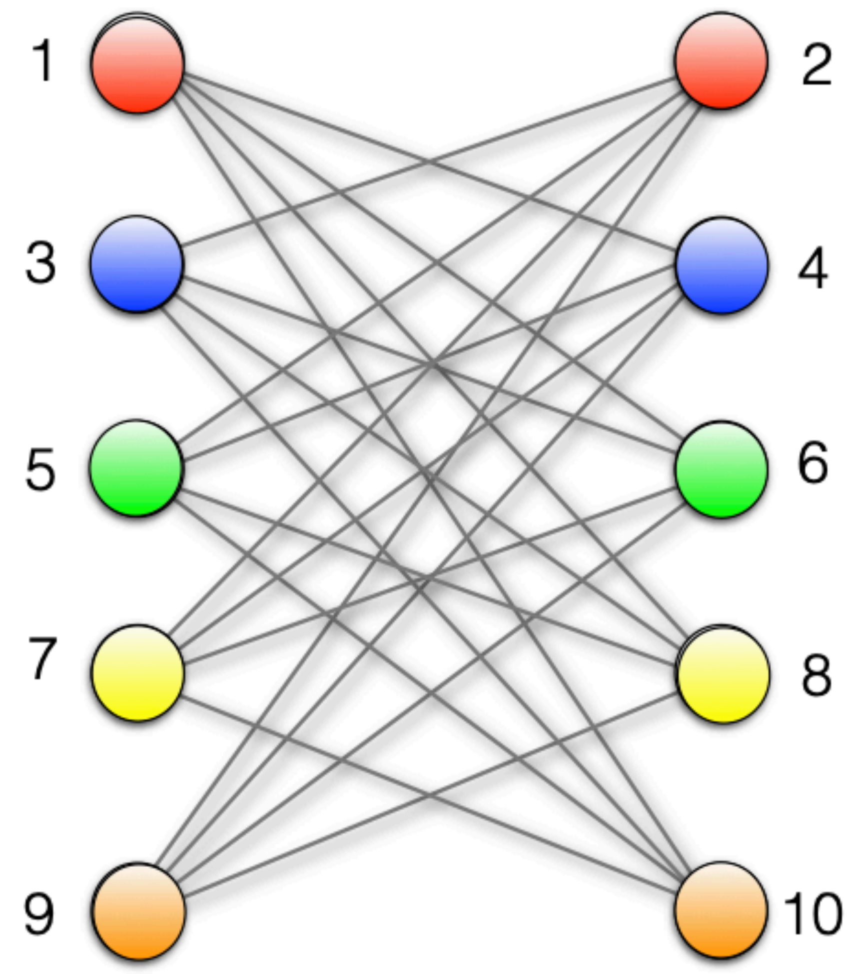
$\underline{k > 2}$: **NP-vollständig**

Farbklassen tauschen:

Falls wir **jeden Block** mit k Farben färben können,

können wir **den ganzen Graphen** mit k Farben färben

Greedy-Färbung



wähle eine beliebige Reihenfolge der Knoten: $V = \{v_1, v_2, \ldots, v_n\}$

$c[v_1] \leftarrow 1$

for $i = 2$ **to** $i = n$ **do**

$\quad c[v_i] \leftarrow \min\{k \in \mathbb{N} \mid k \neq c(u)$ **für alle** $u \in \mathcal{N}(v_i) \cap \{v_1, \ldots, v_{i-1}\}\}$

Für jede Reihenfolge der Knoten braucht der Greedy-Algo höchstens $\Delta(G) + 1$ viele Farben

Es gibt eine Reihenfolge der Knoten, für die der Greedy-Algo nur $\chi(G)$ viele Farben braucht

Es gibt bipartite Graphen und eine Reihenfolge der Knoten, für die der Greedy-Algo $|V|/2$ viele Farben braucht

Heuristik:

$v_n =$ Knote vom kleinsten Grad. Lösche v_n

$v_{n-1} =$ Knote vom kleinsten Grad im Restgraph. Lösche v_{n-1}. Iteriere

Bemerkungen:

1) Die Heuristik findet immer eine Färbung mit 2 Farben für **Bäume**
2) Die Heuristik findet eine Färbung mit 6 Farben für **planare Graphen**
3) Falls $G = (V, E)$ zusammenhängend und $\exists v \in V : \deg(v) < \Delta(G)$: (Alle Graphen außer reguläre Graphen)
   Die Heuristik liefert Reihenfolge, für die der Greedy-Algo höchstens $\Delta(G)$ Farben braucht

3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben

3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

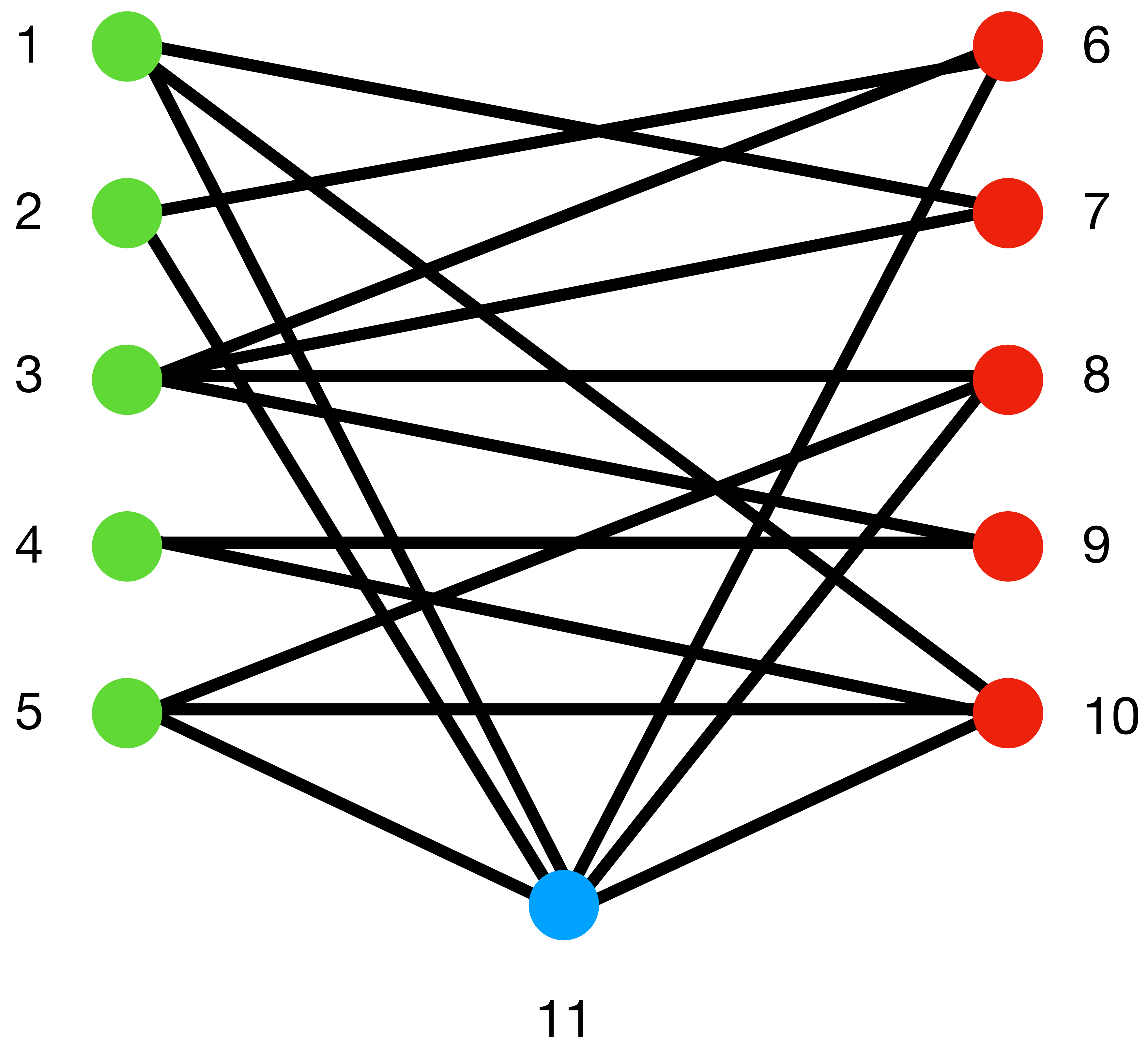
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

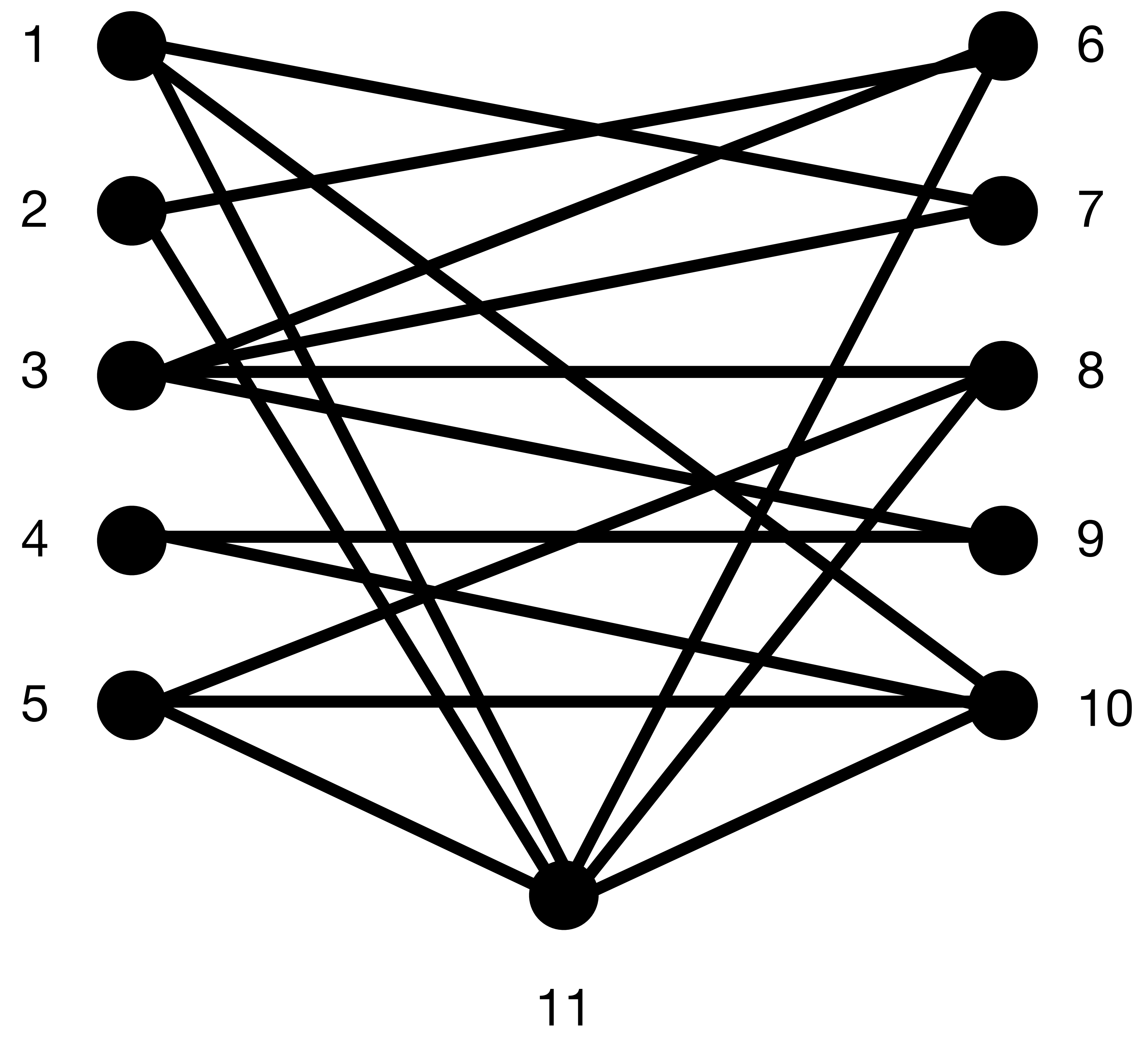
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

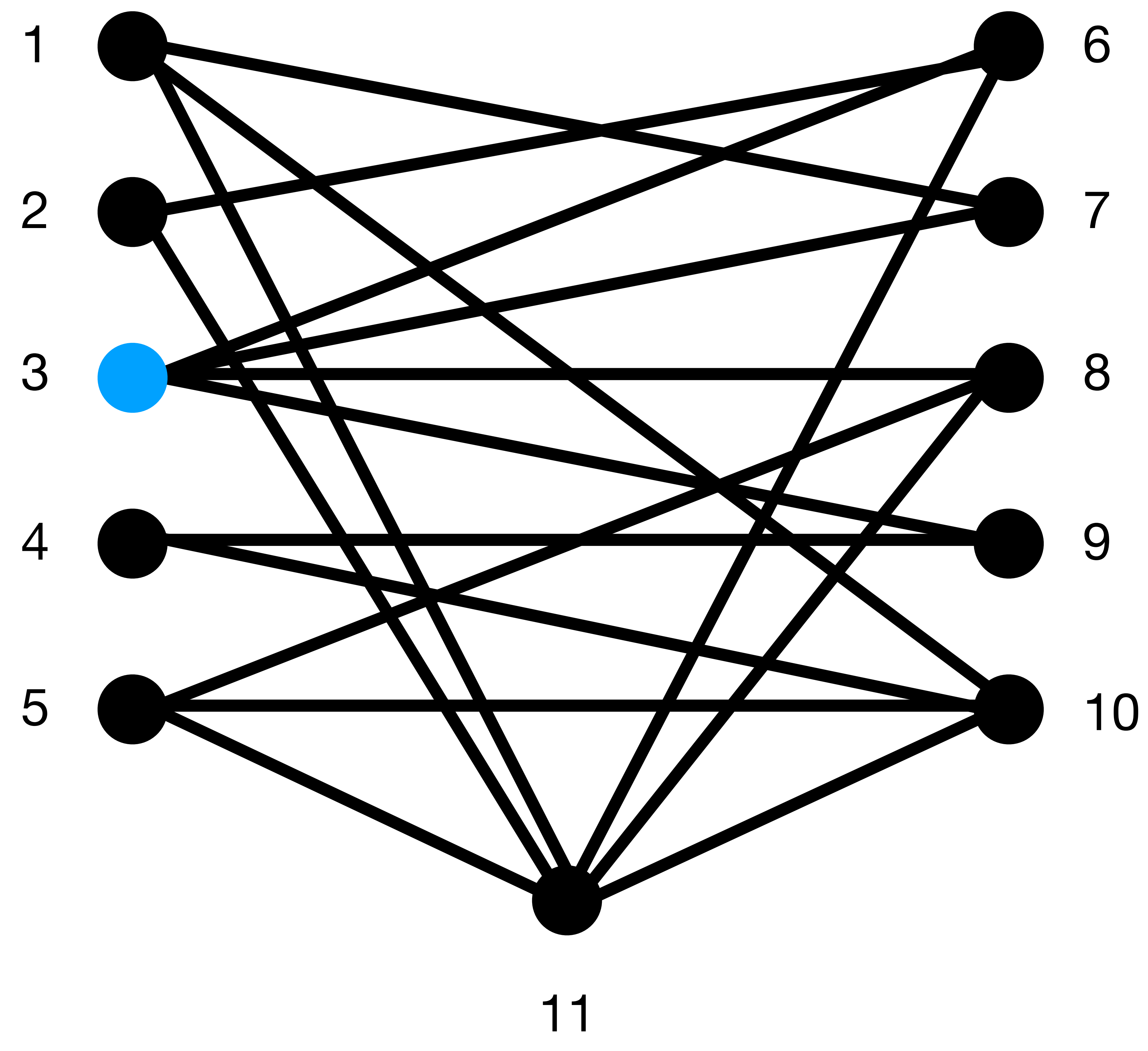
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

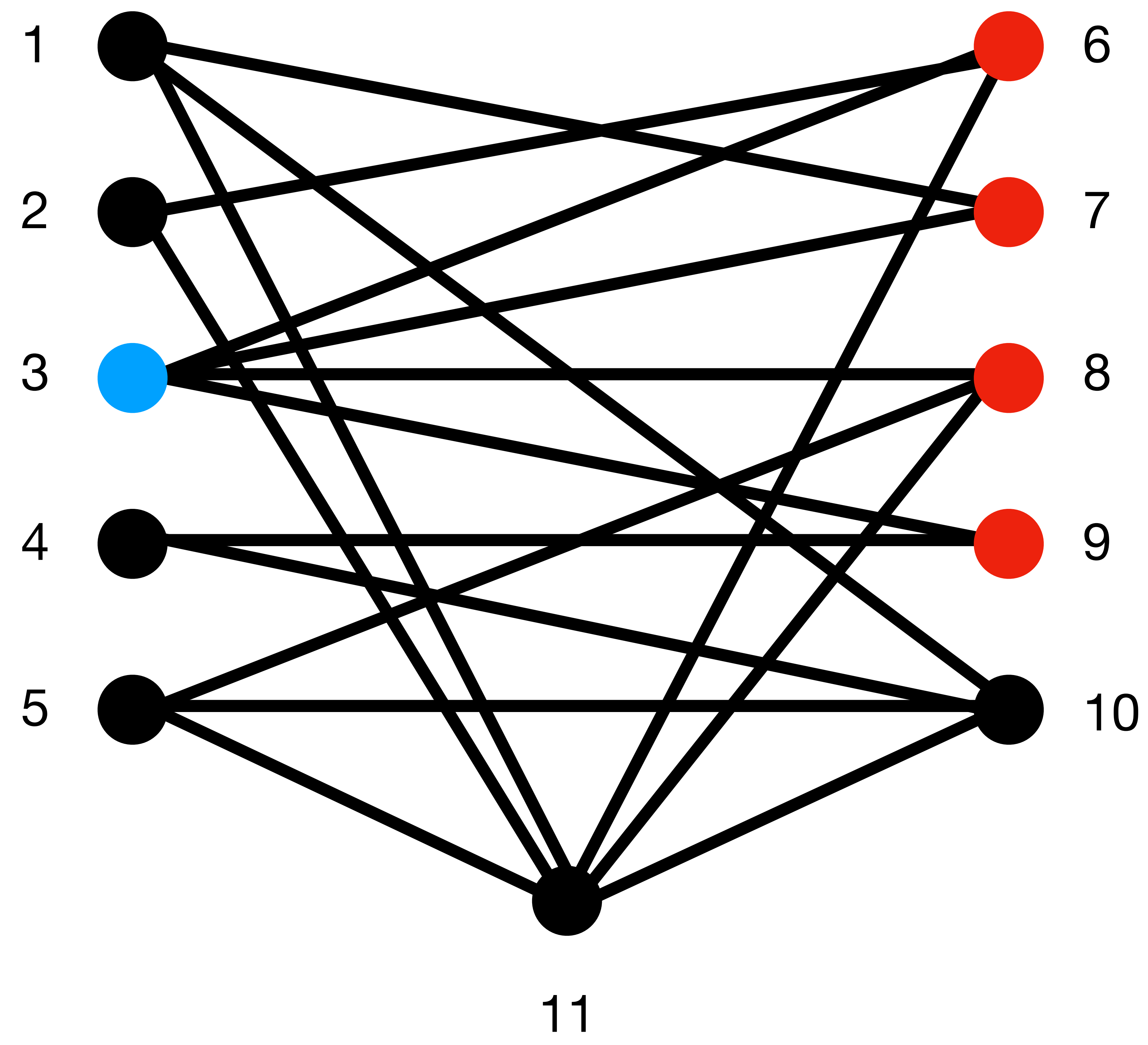
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

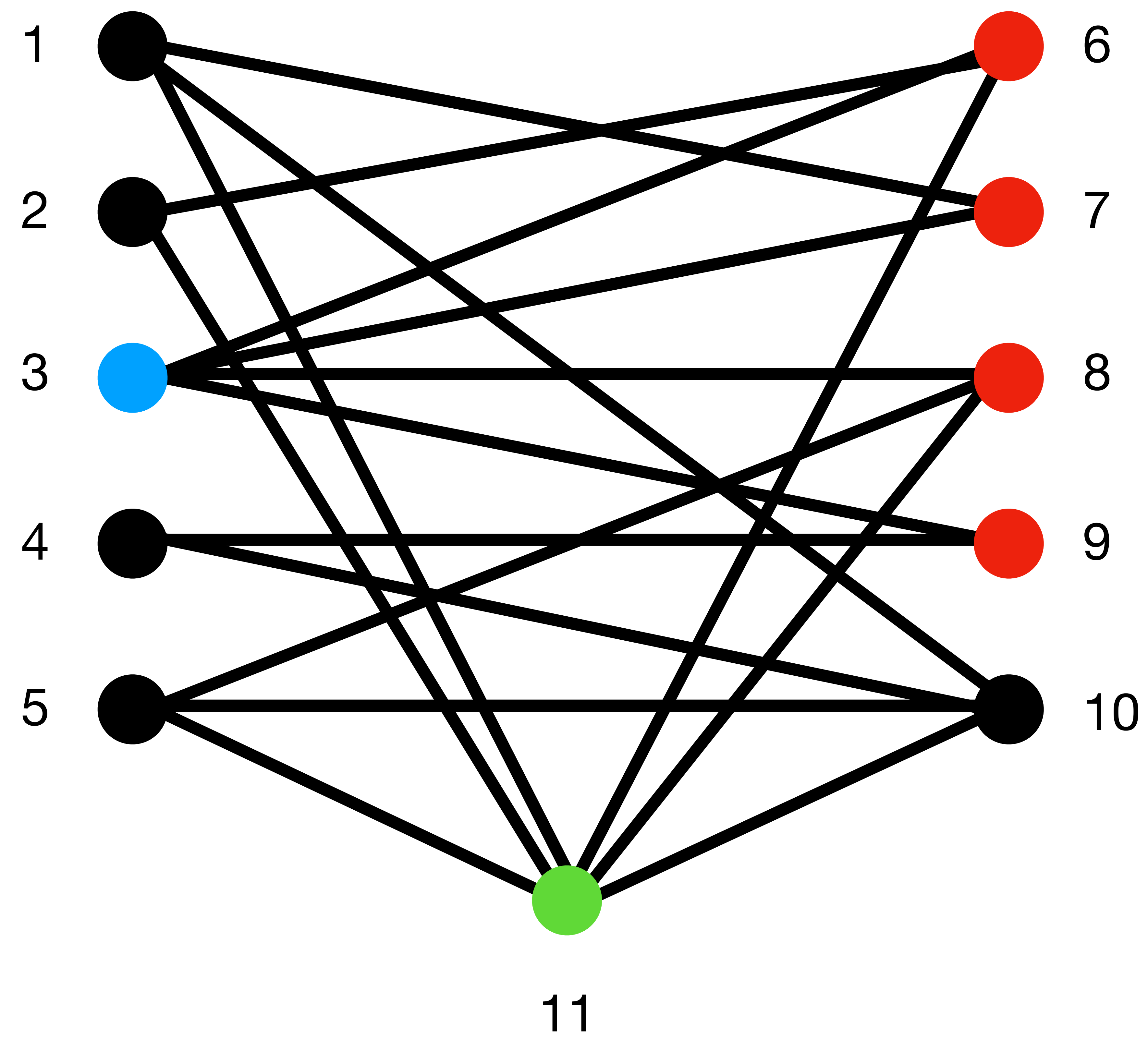
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



3-Färbung

Satz: Einen **3-färbbaren** Graphen kann man in Zeit $O(|V| + |E|)$ mit $O(\sqrt{|V|})$ Farben färben

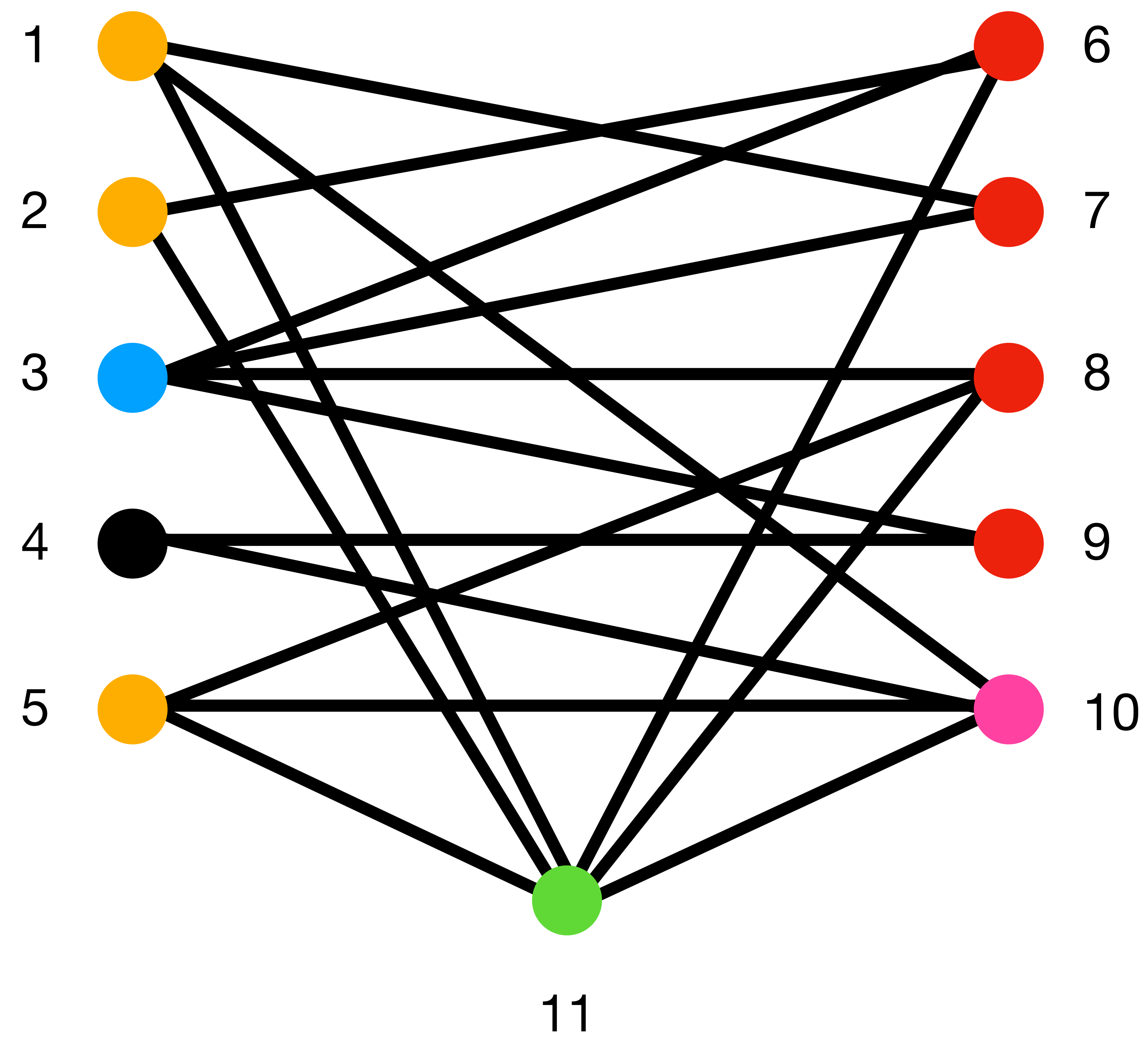
Algorithmus:

while es gibt Knoten v mit $\deg(v) > \sqrt{|V|}$

  färbe v mit neuer Farbe und seine Nachbarn mit 2 weiteren neuen Farben

lösche alle gefärbten Knoten, der Restgraph hat $\Delta(G) \leq \sqrt{|V|}$

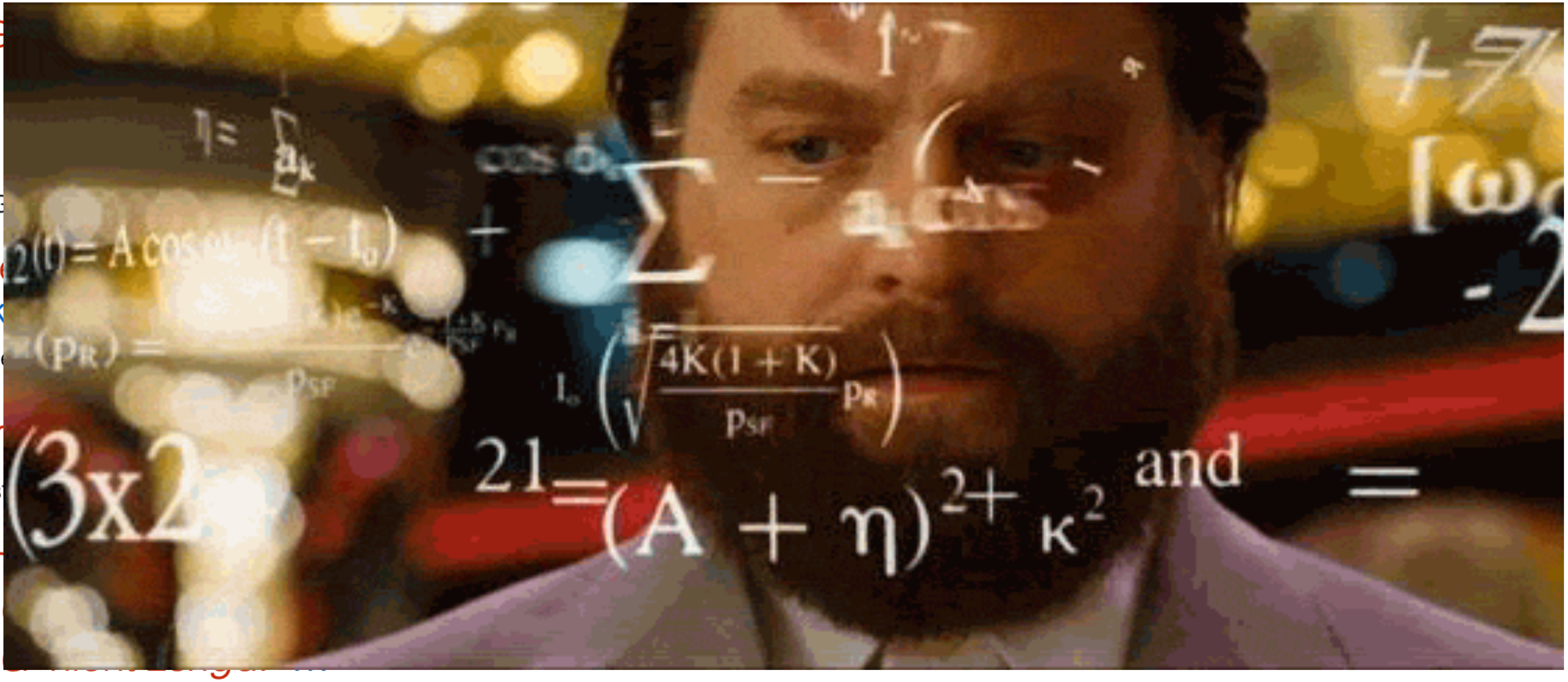
färbe verbleibende Knoten greedy mit $\Delta + 1$ neuen Farben



Satz von Brooks

$G \neq K_n, G \neq C_{2n+1}, G$ zsmhd $\implies G$ kann in $O(|E|)$ mit $\Delta(G)$ Farben gefärbt werden

- Falls Δ(G
  (da G zshgd
- Falls ∃ v
  (geht mit Δ(G
- Falls es e
  Heuristik
  (in allen diese
- Bestimm
  (diese exis
- Betrach
  - *Falls*
  - *Falls*



Wahrscheinlichkeitstheorie

Kombinatorik

	Geordnet	Ungeordnet
Mit Zürücklegen	n^k	$\binom{n+k-1}{k}$
Ohne Zürücklegen	$n^{\underline{k}}$	$\binom{n}{k}$

	Geordnet	Ungeordnet
Mit Zürücklegen	n^k	$\frac{(n+k-1)!}{k!(n-1)!}$
Ohne Zürücklegen	$\frac{n!}{(n-k)!}$	$\frac{n!}{k!(n-k)!}$

Beispiele:

1. Anzahl der verschiedenen bit strings der Länge k
   = 2^k
2. Anzahl Möglichkeiten 11 Spieler aus einer Mannschaft von 22 auszuwählen, wobei die Reihenfolge wichtig ist.
   $$= \frac{22!}{(22-11)!}$$
3. Anzahl Möglichkeiten 3 Kugeln aus 5 verschiedenen Kugelfarben zu ziehen, wenn jede Kugel nach dem Ziehen zurückgelegt wird?
   $$= \binom{5+3-1}{3}$$
4. Anzahl der möglichen Kanten im Graphen
   $$= \binom{n}{2}$$
   → ziehe 2 Elemente aus $[n]$ ohne zurücklegen

Wahrscheinlichkeit - Grundbegriffe

<u>**Diskreter Wahrscheinlichkeitsraum**</u>: $(\Omega, \Pr[\,\cdot\,])$

Ergebnismenge Ω: Menge von <u>Elementarereignissen</u>

Ereignis E: $E \subseteq \Omega$, d.h. eine Menge von Elementarereignissen

Komplementärereignis $\overline{E}$ von E: $\overline{E} := \Omega \backslash E$

1. $\forall \omega \in \Omega : 0 \leq \Pr[\omega] \leq 1$
2. $\sum_{\omega \in \Omega} \Pr[\omega] = 1$
3. $\forall E \subseteq \Omega : \Pr[E] = \sum_{\omega \in E} \Pr[\omega]$

<u>**Laplace-Raum**</u>: Endlicher W-Raum in dem alle Elementarereignisse gleich wahrscheinlich sind

$$\forall \omega \in \Omega : \Pr[\omega] = \frac{1}{|\Omega|}, \Pr[E] = \frac{|E|}{|\Omega|}$$

<u>**Bedingte Wahrscheinlichkeit:**</u>

Für Ereignisse A, B s.d. $\Pr[B] > 0$, $\Pr[A \mid B] = \frac{\Pr[A \cap B]}{\Pr[B]}$

$$\Pr[A \cap B] = \Pr[A \mid B] \cdot \Pr[B] = \Pr[B \mid A] \cdot \Pr[A]$$

Wahrscheinlichkeit - Lemmas

Additionssatz

Für paarweise disjunkte Ereignisse $A_1, \ldots, A_n$

$$\Pr\left[\bigcup_{i=1}^{n} A_i\right] = \sum_{i=1}^{n} \Pr[A_i]$$

Boolsche Ungleichung

$$\Pr\left[\bigcup_{i=1}^{n} A_i\right] \leq \sum_{i=1}^{n} \Pr[A_i]$$

Siebformel

$$\Pr\left[\bigcup_{i=1}^{n} A_i\right] = \sum_{k=1}^{n} \left((-1)^{k+1} \sum_{S \subseteq [n], |S|=k} \Pr\left[\bigcap_{i \in S} A_i\right] \right)$$

1) $\Pr[\emptyset] = 0, \Pr[\Omega] = 1$
2) $0 \leq \Pr[A] \leq 1$
3) $\Pr[\overline{A}] = 1 - \Pr[A]$
4) $A \subseteq B \Longrightarrow \Pr[A] \leq \Pr[B]$

Bedingte Wahrscheinlichkeiten

Multiplikationssatz: Für $A_1, ..., A_n$, falls $\Pr[A_1 \cap A_2 \cap \cdots \cap A_n] > 0$,

$$\Pr[A_1 \cap \cdots \cap A_n] = \Pr[A_1] \cdot \Pr[A_2 \mid A_1] \cdot \Pr[A_3 \mid A_1 \cap A_2] \cdots \Pr[A_n \mid A_1 \cap \cdots \cap A_{n-1}]$$

Satz der totalen Wahrscheinlichkeit: Für paarweise disjunkte Ereignisse $A_1, ..., A_n$ und B, s.d. $B \subseteq A_1 \cup \cdots \cup A_n$,

$$\Pr[B] = \sum_{i=1}^{n} \underbrace{\Pr[B \mid A_i] \cdot \Pr[A_i]}_{=\Pr[B \cap A_i]}$$

Satz von Bayes: Für paarweise disjunkte Ereignisse $A_1, ..., A_n$ und B, s.d. $B \subseteq A_1 \cup \cdots \cup A_n$, $\Pr[B] > 0$,

$$\forall i \in [n] : \Pr[A_i \mid B] = \frac{\Pr[A_i \cap B]}{\Pr[B]} = \frac{\Pr[B \mid A_i] \cdot \Pr[A_i]}{\sum_{j=1}^{n} \Pr[B \mid A_j] \cdot \Pr[A_j]}$$