Fünffärbung planarer Graphen in Linearzeit

Manfred Kufleitner

Terminplanung (Teil 1)

- 7 Praktika werden angeboten. Jeder Studierende muss 2 wählen und absolvieren.

Terminplanung (Teil 1)

- 7 Praktika werden angeboten. Jeder Studierende muss 2 wählen und absolvieren.
- Folgende Paare werden von mindestens einem Studierenden gewählt:

  $(1,2)$ $(1,3)$ $(1,4)$ $(1,5)$ $(1,7)$ $(2,3)$ $(2,4)$ $(2,5)$
  $(2,7)$ $(3,4)$ $(3,6)$ $(4,5)$ $(4,6)$ $(5,7)$ $(6,7)$

Terminplanung (Teil 1)

- 7 Praktika werden angeboten. Jeder Studierende muss 2 wählen und absolvieren.
- Folgende Paare werden von mindestens einem Studierenden gewählt:

  $(1,2)$ $(1,3)$ $(1,4)$ $(1,5)$ $(1,7)$ $(2,3)$ $(2,4)$ $(2,5)$
  $(2,7)$ $(3,4)$ $(3,6)$ $(4,5)$ $(4,6)$ $(5,7)$ $(6,7)$
- Zwei Praktika dürfen gleichzeitig stattfinden, wenn kein Studierender an beiden teilnimmt.

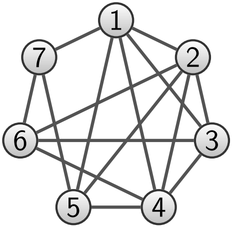
Terminplanung (Teil 1)

- 7 Praktika werden angeboten. Jeder Studierende muss 2 wählen und absolvieren.
- Folgende Paare werden von mindestens einem Studierenden gewählt:

  $(1,2)$ $(1,3)$ $(1,4)$ $(1,5)$ $(1,7)$ $(2,3)$ $(2,4)$ $(2,5)$
  $(2,7)$ $(3,4)$ $(3,6)$ $(4,5)$ $(4,6)$ $(5,7)$ $(6,7)$
- Zwei Praktika dürfen gleichzeitig stattfinden, wenn kein Studierender an beiden teilnimmt.
- Wieviele Termine werden benötigt?

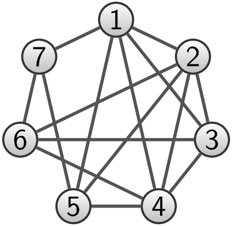
Terminplanung (Teil 2)

- Der folgende Graph veranschaulicht den Sachverhalt:



Terminplanung (Teil 2)

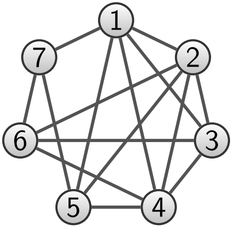
- Der folgende Graph veranschaulicht den Sachverhalt:



- Eine Planung muss berücksichtigen, dass keine über eine Kante verbundenen Praktika zur selben Zeit stattfinden.

Terminplanung (Teil 2)

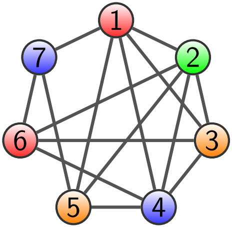
- Der folgende Graph veranschaulicht den Sachverhalt:



- Eine Planung muss berücksichtigen, dass keine über eine Kante verbundenen Praktika zur selben Zeit stattfinden.
- Färbung von Knoten (Praktika) mit Terminen, sodass keine benachbarten Knoten die gleiche Farbe haben

Terminplanung (Teil 2)

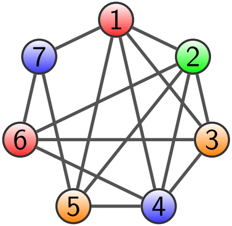
- Der folgende Graph veranschaulicht den Sachverhalt:



- Eine Planung muss berücksichtigen, dass keine über eine Kante verbundenen Praktika zur selben Zeit stattfinden.
- Färbung von Knoten (Praktika) mit Terminen, sodass keine benachbarten Knoten die gleiche Farbe haben
- Färbung bei 4 möglichen Terminen (rot, grün, orange, blau)

Terminplanung (Teil 2)

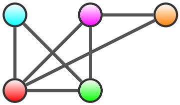
- Der folgende Graph veranschaulicht den Sachverhalt:



- Eine Planung muss berücksichtigen, dass keine über eine Kante verbundenen Praktika zur selben Zeit stattfinden.
- Färbung von Knoten (Praktika) mit Terminen, sodass keine benachbarten Knoten die gleiche Farbe haben
- Färbung bei 4 möglichen Terminen (rot, grün, orange, blau)
- Es gibt keine Färbung mit weniger als 4 Farben.

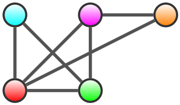
Planare Graphen

- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



Planare Graphen

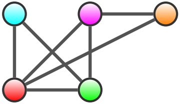
- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x

Planare Graphen

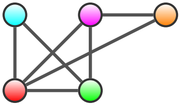
- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x
- $\sum_{x \in V} d_x = 2m$

Planare Graphen

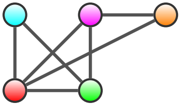
- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



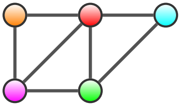
- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x
- $\sum_{x \in V} d_x = 2m$
- Ein Graph ist planar, wenn man ihn ohne Kantenüberschneidungen auf der Ebene zeichnen kann.

Planare Graphen

- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten

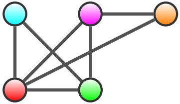


- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x
- $\sum_{x \in V} d_x = 2m$
- Ein Graph ist planar, wenn man ihn ohne Kantenüberschneidungen auf der Ebene zeichnen kann.

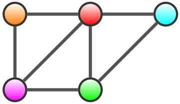


Planare Graphen

- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



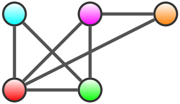
- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x
- $\sum_{x \in V} d_x = 2m$
- Ein Graph ist planar, wenn man ihn ohne Kantenüberschneidungen auf der Ebene zeichnen kann.



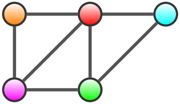
- Wenn G planar mit $n \geq 3$, dann $m \leq 3n - 6$.

Planare Graphen

- Graph $G = (V, E)$, ungerichtet, keine Schlingen, keine Mehrfachkanten



- $n = |V|$, $m = |E|$, $d_x =$ Grad von Knoten x
- $\sum_{x \in V} d_x = 2m$
- Ein Graph ist planar, wenn man ihn ohne Kantenüberschneidungen auf der Ebene zeichnen kann.



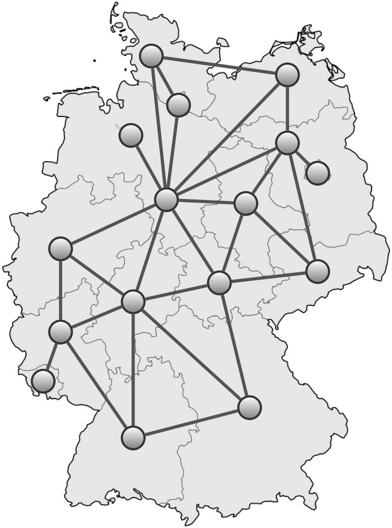
- Wenn G planar mit $n \geq 3$, dann $m \leq 3n - 6$.
- Wenn G planar, dann existiert Knoten x mit $d_x \leq 5$. (Sonst wäre $6n \leq \sum_x d_x = 2m \leq 6n - 12$.)

Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten

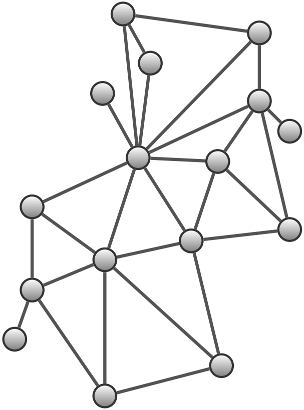
Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten



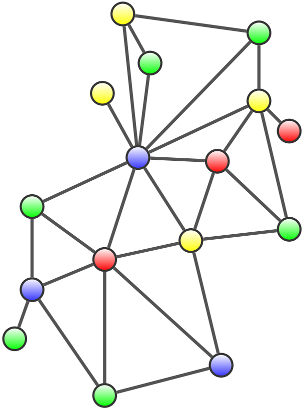
Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten



Färbbarkeit planarer Graphen

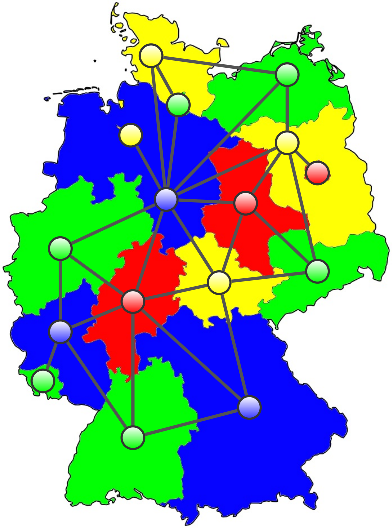
- Historisch: Färbung von Landkarten
- Vierfarbensatz: Planare Graphen sind 4-färbbar.



Färbbarkeit planarer Graphen

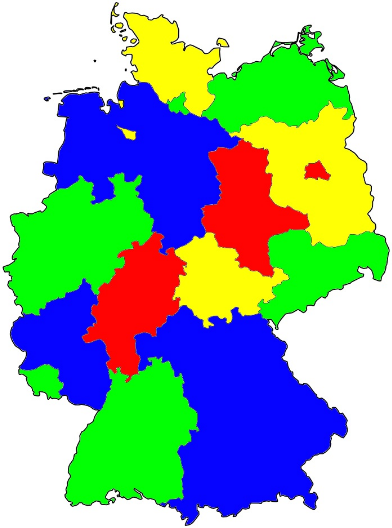
- Historisch: Färbung von Landkarten

- Vierfarbensatz: Planare Graphen sind 4-färbbar.



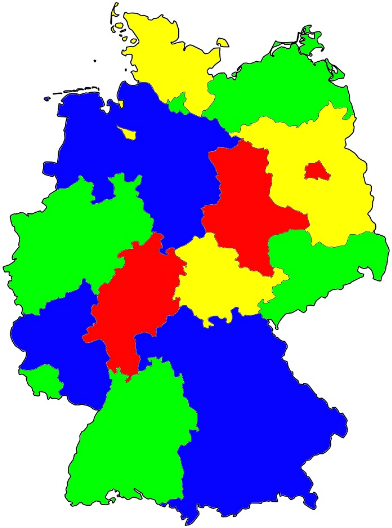
Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten
- Vierfarbensatz: Planare Graphen sind 4-färbbar.



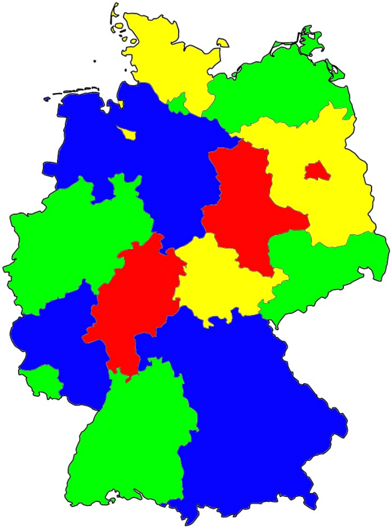
Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten
- Vierfarbensatz: Planare Graphen sind 4-färbbar.
- Appel, Haken 1976: 1936 Fälle



Färbbarkeit planarer Graphen

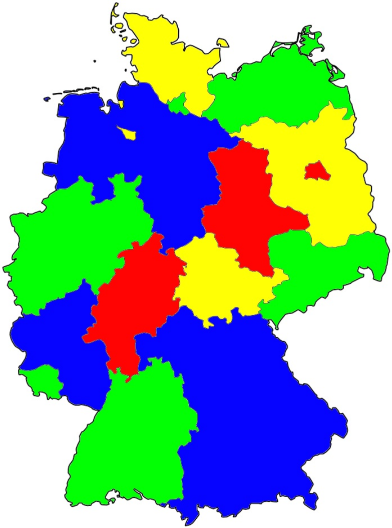
- Historisch: Färbung von Landkarten
- Vierfarbensatz: Planare Graphen sind 4-färbbar.
- Appel, Haken 1976: 1936 Fälle
- Appel, Haken 1977: 1476 Fälle



Färbbarkeit planarer Graphen

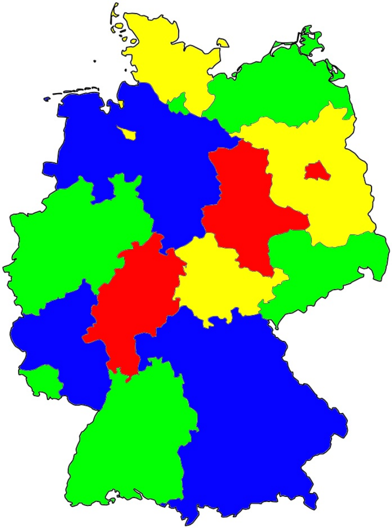
- Historisch: Färbung von Landkarten

- Vierfarbensatz: Planare Graphen sind 4-färbbar.
- Appel, Haken 1976: 1936 Fälle
- Appel, Haken 1977: 1476 Fälle
- Robertson, Sanders, Seymour, Thomas 1996: 633 Fälle



Färbbarkeit planarer Graphen

- Historisch: Färbung von Landkarten
- Vierfarbensatz: Planare Graphen sind 4-färbbar.
- Appel, Haken 1976: 1936 Fälle
- Appel, Haken 1977: 1476 Fälle
- Robertson, Sanders, Seymour, Thomas 1996: 633 Fälle
- $\mathcal{O}(n^2)$ Algorithmus zur Berechnung einer 4-Färbung



6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

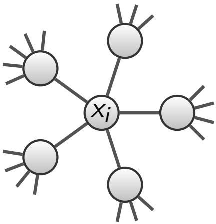
  for $i = 1, \ldots, n$ **do**

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;



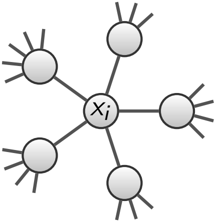
6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;

  Entferne x_i;



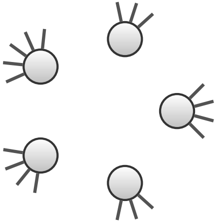
6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;

  Entferne x_i;



6-Färbung planarer Graphen

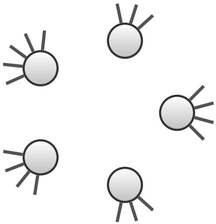
- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;

  Entferne x_i;

  Update der Nachbarn von x_i;



6-Färbung planarer Graphen

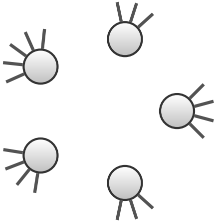
- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;

  Entferne x_i;

  Update der Nachbarn von x_i;

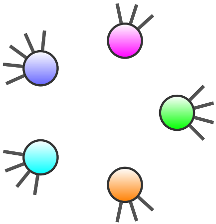


- Schritt 3: Färbe $x_n, \ldots, x_1$ nacheinander mit freier Farbe.

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  Entferne x_i;
  Update der Nachbarn von x_i;

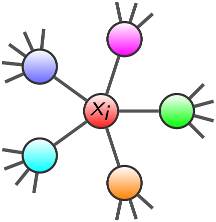


- Schritt 3: Färbe $x_n, \ldots, x_1$ nacheinander mit freier Farbe.

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2:

  for $i = 1, \ldots, n$ **do**
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  Entferne x_i;
  Update der Nachbarn von x_i;

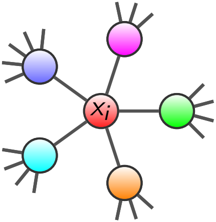


- Schritt 3: Färbe $x_n, \ldots, x_1$ nacheinander mit freier Farbe.

6-Färbung planarer Graphen

- Eingabe: planarer Graph $G = (V, E)$ als Adjazenzliste
- Schritt 1: Liste L_i enthält die Knoten mit Grad i $\quad | \sum_x d_x$
- Schritt 2:

  for $i = 1, \ldots, n$ **do**
  $\quad x_i \leftarrow$ Knoten aus $\underline{L_0 \cup \cdots \cup L_5}$;
  $\quad$Entferne x_i;
  $\quad$Update der Nachbarn von x_i;



- Schritt 3: Färbe $x_n, \ldots, x_1$ nacheinander mit freier Farbe. $\quad 5n$
- Alle Schritte sind linear.

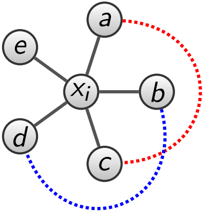
5-Färbung planarer Graphen: Idee

Bei höchstens vier Nachbarn ist eine der fünf Farben für x_i frei.

5-Färbung planarer Graphen: Idee

Bei höchstens vier Nachbarn ist eine der fünf Farben für x_i frei.

Fünf Nachbarn:



Kanten *ac* und *bd* sind nicht gleichzeitig möglich.

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i

5-Färbung planarer Graphen: Algorithmus

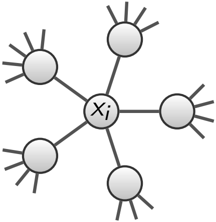
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

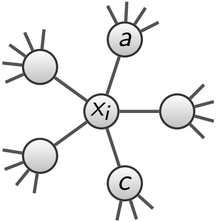
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

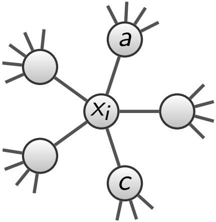
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

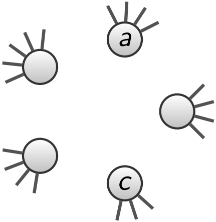
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

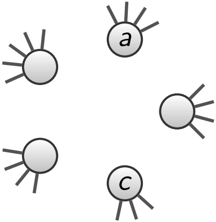
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

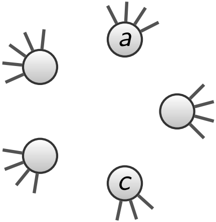
$x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
if $d_{x_i} < 5$ **then**
  Entferne x_i;
else
  Finde $a, c \in N(x_i)$ mit $ac \notin E$;
  Entferne x_i;
  $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

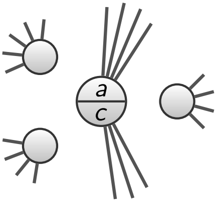
$x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
if $d_{x_i} < 5$ **then**
    Entferne x_i;
else
    Finde $a, c \in N(x_i)$ mit $ac \notin E$;
    Entferne x_i;
    $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
    Verschmelze a, c zu c;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

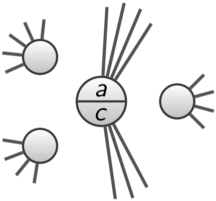
$x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
if $d_{x_i} < 5$ **then**
    Entferne x_i;
else
    Finde $a, c \in N(x_i)$ mit $ac \notin E$;
    Entferne x_i;
    $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
    Verschmelze a, c zu c;



5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;
      $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
      Verschmelze a, c zu c;

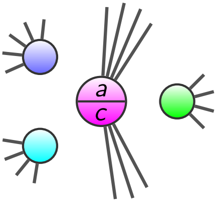


- Schritt 3: Färbe $x_n, \ldots, x_1$;

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \dots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \dots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;
      $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
      Verschmelze a, c zu c;

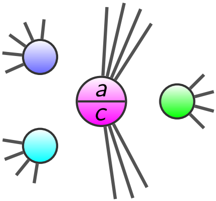


- Schritt 3: Färbe $x_n, \dots, x_1$;

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;
      $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
      Verschmelze a, c zu c;

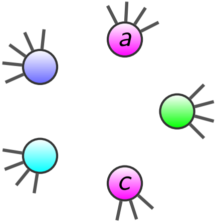


- Schritt 3: Färbe $x_n, \ldots, x_1$;
  bei Verweis $x_{i+1} \rightsquigarrow c$, verwende für x_{i+1} die Farbe von c.

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;
      $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
      Verschmelze a, c zu c;

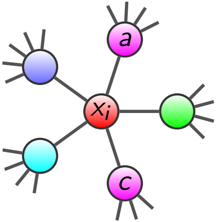


- Schritt 3: Färbe $x_n, \ldots, x_1$;
  bei Verweis $x_{i+1} \rightsquigarrow c$, verwende für x_{i+1} die Farbe von c.

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_5$;
  if $d_{x_i} < 5$ **then**
  Entferne x_i;
  else
  Finde $a, c \in N(x_i)$ mit $ac \notin E$;
  Entferne x_i;
  $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
  Verschmelze a, c zu c;

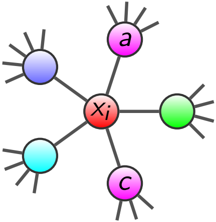


- Schritt 3: Färbe $x_n, \ldots, x_1$;
  bei Verweis $x_{i+1} \rightsquigarrow c$, verwende für x_{i+1} die Farbe von c.

5-Färbung planarer Graphen: Algorithmus

- Schritt 1: Liste L_i enthält die Knoten mit Grad i
- Schritt 2: Schleife über $i = 1, \dots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \dots \cup L_5$;
  if $d_{x_i} < 5$ **then**
  Entferne x_i;
  else
  Nicht konst. → Finde $a, c \in N(x_i)$ mit $ac \notin E$;
  Entferne x_i;
  $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
  Verschmelze a, c zu c;



- Schritt 3: Färbe $x_n, \dots, x_1$;
  bei Verweis $x_{i+1} \rightsquigarrow c$, verwende für x_{i+1} die Farbe von c.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned}
\textstyle\sum_x (30 - 5d_x) &= 30n - 5\textstyle\sum_x d_x \\
&= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\
&\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6
\end{aligned}$$

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned} \textstyle\sum_x (30 - 5d_x) &= 30n - 5\textstyle\sum_x d_x \\ &= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\ &\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6 \end{aligned}$$

- Jeder Knoten mit Grad 5 gibt jedem seiner Nachbarn 1 Euro.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned} \textstyle\sum_x (30 - 5d_x) &= 30n - 5\textstyle\sum_x d_x \\ &= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\ &\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6 \end{aligned}$$

- Jeder Knoten mit Grad 5 gibt jedem seiner Nachbarn 1 Euro.
- Das Gesamtguthaben ist unverändert größer als 0.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned} \textstyle\sum_x (30 - 5d_x) &= 30n - 5 \textstyle\sum_x d_x \\ &= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\ &\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6 \end{aligned}$$

- Jeder Knoten mit Grad 5 gibt jedem seiner Nachbarn 1 Euro.
- Das Gesamtguthaben ist unverändert größer als 0.
- Also gibt es einen Knoten y mit positivem Kontostand K.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned}\textstyle\sum_x (30 - 5d_x) &= 30n - 5\textstyle\sum_x d_x \\ &= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\ &\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6\end{aligned}$$

- Jeder Knoten mit Grad 5 gibt jedem seiner Nachbarn 1 Euro.
- Das Gesamtguthaben ist unverändert größer als 0.
- Also gibt es einen Knoten y mit positivem Kontostand K.
- $K \leq 30 - 5d_y + d_y$, da jeder Nachbar maximal 1 Euro gibt.

Der Satz von Wernicke

Satz (Wernicke, 1904): Wenn G planar ist mit Minimalgrad 5, dann existiert $xy \in E$ mit $d_x = 5$ und $d_y \leq 7$.

Beweis:

- Jeder Knoten x habe einen Kontostand von $30 - 5\,d_x$ Euro.
- Gesamtguthaben:

$$\begin{aligned}
\textstyle\sum_x (30 - 5d_x) &= 30n - 5 \textstyle\sum_x d_x \\
&= 30n - 10m && \text{da } \textstyle\sum_x d_x = 2m \\
&\geq 30n - 30n + 60 = 60 > 0 && \text{da } m \leq 3n - 6
\end{aligned}$$

- Jeder Knoten mit Grad 5 gibt jedem seiner Nachbarn 1 Euro.
- Das Gesamtguthaben ist unverändert größer als 0.
- Also gibt es einen Knoten y mit positivem Kontostand K.
- $K \leq 30 - 5d_y + d_y$, da jeder Nachbar maximal 1 Euro gibt.
- Aus $0 < K \leq 30 - 5d_y + d_y = 30 - 4d_y$ folgt $d_y \leq 7$. $\square$

5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet

5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i

5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L_5' \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7

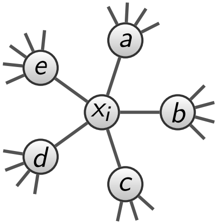
5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L_5' \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7
- Schritt 2: Schleife über $i = 1, \ldots, n$
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_4 \cup L_5'$;

5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L'_5 \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7
- Schritt 2: Schleife über $i = 1, \ldots, n$

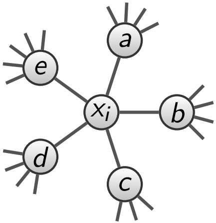
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_4 \cup L'_5$;
  if $d_{x_i} < 5$ **then**
  Entferne x_i;
  else



5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L_5' \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7
- Schritt 2: Schleife über $i = 1, \dots, n$

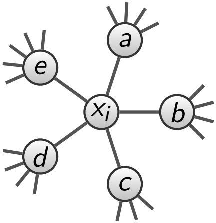
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_4 \cup L_5'$;
  if $d_{x_i} < 5$ **then**
  Entferne x_i;
  else
  Finde $a, c \in N(x_i)$ mit $ac \notin E$;



5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L'_5 \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7
- Schritt 2: Schleife über $i = 1, \ldots, n$

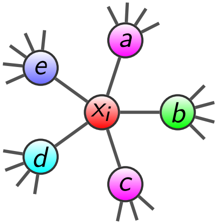
  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_4 \cup L'_5$;
  if $d_{x_i} < 5$ **then**
      Entferne x_i;
  else
      Finde $a, c \in N(x_i)$ mit $ac \notin E$;
      Entferne x_i;
      $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
      Verschmelze a, c zu c;



5-Färbung planarer Graphen: Linearzeit-Algorithmus

- Eingabe: Adjazenzliste, Nachbarn im Uhrzeigersinn angeordnet
- Schritt 1: Liste L_i enthält die Knoten mit Grad i
  $L'_5 \leftarrow$ Knoten aus L_5 mit einem Nachbarn von Grad ≤ 7
- Schritt 2: Schleife über $i = 1, \ldots, n$

  $x_i \leftarrow$ Knoten aus $L_0 \cup \cdots \cup L_4 \cup L'_5$;
  if $d_{x_i} < 5$ **then**
  Entferne x_i;
  else
  Finde $a, c \in N(x_i)$ mit $ac \notin E$;
  Entferne x_i;
  $x_{i+1} \leftarrow a$; Verweis $x_{i+1} \rightsquigarrow c$;
  Verschmelze a, c zu c;



- Schritt 3: Färbe $x_n, \ldots, x_1$;
  bei Verweis $x_{i+1} \rightsquigarrow c$, verwende für x_{i+1} die Farbe von c.

Danke für Ihre Aufmerksamkeit!

Danke für Ihre Aufmerksamkeit!