Kartenfärben

Landkarten werden normalerweise koloriert, um einzelne Gebiete gut unterscheiden zu können. Dabei dürfen benachbarte Gebiete nicht in derselben Farbe gefärbt werden. Es sollen dabei möglichst wenige Farben verwendet werden.

(A1)

Färbe die Karte der Bundesländer nach der beschriebenen Regel.

[1)]

Modellierung

Die Ausgangssituation soll nun als Graph modelliert werden. Dabei stehen die Knoten für die Gebiete der Landkarte, zwei Knoten haben eine gemeinsame Kante, wenn Sie auf der Karte eine gemeinsame Grenzlinie haben.

- https://www.info-bw.de/

Graphenfärbe-Problem: Geben ist ein Graph. Färbe die Knoten des Graphen so, dass keine durch eine Kante verbundene Knoten die gleiche Farbe haben.

Varianten:

- Verwende dabei möglichst wenige Farben.
- Ist es möglich, den Graphen mit k Farben zu färben?

Weiterführende Fragen & Aufgaben

Für die Kolorierung von Graphen gelten folgende Sätze:

- Graphen, die sich mit einer Farbe färben lassen, haben keine Kante außer Schleifen.
- Ein bipartiter[2)] Graph lässt sich mit zwei Farben färben.
- Ein vollständiger Graph mit n Knoten benötigt n Farben.
- Ein Graph mit einer Clique[3)] aus m Knoten benötigt mindestens m Farben.

(A2)

Begründe die obigen Aussagen.

Lösung

- Sobald eine Kante vorhanden ist, sind zwei Knoten verbunden. Wenn dies verschiedene Knoten sind (also keine Schleife), dann dürfen diese nicht die selbe Farbe haben und es werden mindestens zwei Farben benötigt.
- Wenn der Graph bipartit ist, dann zerfällt er in zwei Teilmenge der Knoten, die untereinander überhaupt nicht verbunden sind. Damit kann jede Teilmenge in einer einzigen Farbe eingefärbt werden.
- Bei einem vollständigen Graphen ist jeder Knoten mit allen anderen verbunden. Daher muss jeder Knoten eine eigene Farbe haben.
- In einer Clique sind alle Knoten untereinander verbunden. Daher muss jeder Knoten der Clique eine eigene Farbe bekommen.

(A3)

Beschreibe eine Situation (Landkarte incl. Reihenfolge der Länder), in der der Greedy-Algorithmus mehr als 4 Farben erfordert.

Lösungsvorschlag

Werden die Knoten in der angegebenen Reihenfolge bearbeitet, findet der Greedy-Algorithmus (Farbreihenfolge: blau-rot-grün-gelb-lila) die gezeigte Lösung mit 5 Farben, da das Land 2 ungeschickterweise mit rot statt grün gefärbt wurden.

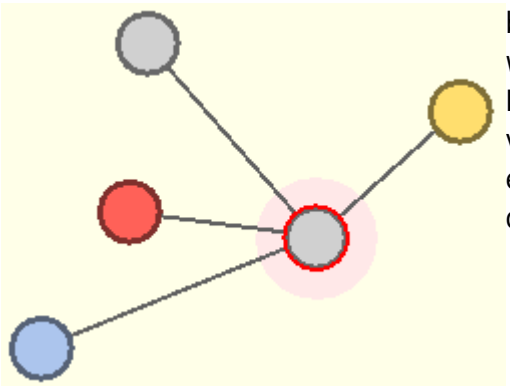
Algorithmus

Beschreibung

Der hier beschriebene Algorithmus findet nicht die perfekte Lösung, d.h. die minimale Anzahl an Farben, aber eine gute Näherungslösung. Er arbeitet dabei nach dem Greedy-Verfahren, er wählt für ein Land die momentan am besten erscheinende Lösung.

Zunächst wird eine Reihenfolge festgelegt, in der die Farben verwendet werden sollen.

z.B. Rot - Blau - Grün - Gelb - Lila - Orange - Braun (es müssen ausreichend viele Farben sein) Dann betrachtet man der Reihe nach alle Knoten. Für jeden Knoten wird dann Folgendes gemacht: Man schaut jeden der Nachbarknoten an und merkt sich, dass seine Farbe schon verwendet wurde. Dann wählt man aus der Liste der Farben die erste noch nicht benutzte Farbe aus und färbt den Knoten in dieser Farbe.



z.B. Der rot umrandete Knoten ist aktuell an der Reihe. Alle Nachbarknoten werden betrachtet und

ihre Farben ermittelt.

Farbe	Rot	Blau	Grün	Gelb	Lila	Orange	Braun
Benutzt	ja	ja	nein	ja	nein	nein	nein

Die erste noch nicht benutzte Farbe ist grün. Daher wird der Knoten grün gefärbt.

Pseudocode

Pseudocode

```
Kartenfärbung:
Wiederhole für jeden Knoten k des Graphen
  Wiederhole für jede Farbe der Farbliste
     Setze die Farbe auf "unbenutzt"
  Ende-Wiederhole
  Wiederhole für jeden Nachbarknoten n von k
     Betrachte die Farbe des Knoten n
     Setze diese Farbe auf "benutzt"
  Ende-Wiederhole
  Wiederhole für jede Farbe der Farbliste
     Falls die Farbe "unbenutzt" ist
       Färbe den Knoten k mit dieser Farbe
       Brich die Schleife ab
     Ende-Falls
  Ende-Wiederhole

Ende-Wiederhole
```

Beispielimplementation im Graphentester

Beispielimplementation

```
List<Knoten> knoten = g.getAlleKnoten();
for (Knoten aktuellerKnoten: knoten) {
  boolean[] farbenliste = new boolean[g.getAnzahlKnoten()+1];
  for (int i=0; i < farbenliste.length; i++){
    farbenliste[i]=false;
  }

  List<Knoten> nachbarn = g.getNachbarknoten(aktuellerKnoten);
  for (Knoten k : nachbarn){
    farbenliste[k.getFarbe()]=true;
  }
```

```
  for (int i=1; i<farbenliste.length; i++){
    if (!farbenliste[i]) {
      aktuellerKnoten.setFarbe(i);
      break;
    }
  }
}
```

1)

Bundesländer in Deutschland, Stefan-Xp via Wikimedia Commons (Lizenz: CC BY-SA 3.0): https://de.wikipedia.org/wiki/Datei:Germany_blank_map.svg

2)

https://de.wikipedia.org/wiki/Bipartiter_Graph

3)

Wikipedia: Clique

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:graphen:zpg:kartenfaerben:start?rev=1670326414

Last update: **06.12.2022 11:33**