Gruppe $\boxed{17}$
Berechenbarkeit & Komplexität

Philipp Hochmann, 356148
Felix Kiunke, 357322

Blatt 8

Aufgabe 8.1

Fehlt :(

Aufgabe 8.2

(a)

Der linke Graph ist in 2-COLORABILITY enthalten. Eine Lösung ist:
Schwarz: 3, 4, 6
Weiß: 1, 2, 5, 7.

Der rechte Graph ist nicht enthalten. Dieser Graph enthält einen Kreis mit ungerader Knotenanzahl (1, 3, 2, 4, 7). Dieser Kreis ist offensichtlich nicht 2-färbbar, also ist der gesamte Graph ebenfalls nicht färbbar.

(b)

Wir entwerfen folgenden Algorithmus:

- Starte an einem beliebigen ungefärbten Knoten und färbe diesen weiß. Merke, dass der zuletzt eingefärbte Knoten weiß ist.
- Wähle einen beliebigen Nachbarknoten K des gewählten Knotens. Hat dieser dieselbe Farbe, verwirf, denn es gibt offensichtlich keine 2-Färbung. Hat dieser eine andere Farbe, fahre mit einem anderen Nachbarn (sofern vorhanden) fort. Ist K noch ungefärbt, fahre mit K fort.
- Färbe K mit dem „Gegenteil“ der gemerkten Farbe und merke die für K verwendete Farbe.

Diese Tiefensuche muss solange durchgeführt werden, wie es noch ungefärbte Knoten gibt. Wenn alle Knoten eingefärbt sind, ist dies eine gültige 2-Färbung und die Eingabe kann akzeptiert werden.

Korrektheit: Bei Akzeptanz ist die Korrektheit klar: Jeder neu eingefärbte Knoten ist korrekt eingefärbt, da für alle Nachbarn geprüft wird, ob diese schon die selbe Farbe haben und ansonsten verworfen wird. Außerdem werden die noch nicht eingefärbten Nachbarknoten in einer anderen Farbe eingefärbt und „passen“ somit auch.

Die Einfärbung im Algorithmus ist bis auf Umkehrung der Farben eindeutig. In jedem Schritt ist der bisher eingefärbte Teilgraph gültig 2-gefärbt. Wenn nun zwei Nachbarn in Schritt 2 die gleiche Farbe haben, sind sie (da eine Tiefensuche durchgeführt wird) auch anderweitig (über andere Knoten) verbunden. Das bedeutet, dass die Farbe dieser beiden bereits eingefärbten Knoten ebenfalls bis auf Umkehrung eindeutig ist. Damit gibt es also keine andere gültige Färbung, durch die die beiden Nachbarn nicht die gleiche Färbung erreichen. Also ist der Algorithmus auch im „Verwerf-Fall“ korrekt

Laufzeit: Die Tiefensuche ist in $\mathcal{O}(|V| + |E|)$ durchführbar mit V = Knotenmenge und E = Kantenmenge. Da die Kantenmenge durch die Knotenmenge beschränkt ist, ist die Laufzeit also (grob abgeschätzt) $\mathcal{O}(|V|^3)$. Die Überprüfung der Nachbarn ist in $\mathcal{O}(|V|)$, also liegt die Gesamtlaufzeit in $\mathcal{O}(|V|^4)$ und damit in P

Aufgabe 8.3

Im folgenden wird gezeigt, dass es eine Abbildung V und ein Polynom p gibt, sodass gilt:

$$x \in \text{3-Colorability} \iff \exists |y| \leq p(|x|) : V \text{ akzeptiert } y\#x.$$

Ein Graph mit n Knoten kann mit einer Adjazenzmatrix in n^2 Zeichen aus $0, 1$ kodiert werden, d.h. $|x| = \mathcal{O}(n^2)$. Das Zertifikat der Lösung sei eine Liste an Farben in der Reihenfolge der Knoten. Bei 3 Farben kann jede Farbe mit 2 Bits kodiert werden: 00, 11 oder 01. Es gilt daher $|y| = \mathcal{O}(n)$.

$$p(x) = x, \text{ da } x \leq x^2 \quad \forall x \in \mathbb{N}$$

Verifikation der Lösung, die durch y beschrieben wird: Sei $y_n \in 00, 11, 01$ der Farbwert des n-ten Knotens und $x_{i,j} \in 0, 1$ ein Element aus der Adjazenzmatrix.

for $i = 1..n$ do
for $j = 1..n$ do
if $x_{i,j} = 1$ and $i > j$
if $y_i = y_j$ REJECT
end if
end for
end for ACCEPT

Da dieser Algorithmus n^2 viele Schritte benötigt und Lesen aus y mit polynomialem Zeitverlust mit einem zweiten Band simuliert werden kann, ist dies ein Algorithmus in Polynomialzeit.

Aufgabe 8.4

NP $\subseteq$ NP':
Sei L $\in$ NP, d.h. es existiert eine NTM M mit L(M)=L und Polynom p mit $t_M(n) \leq p(n)$.

Man konstruiere eine NTM M', welche die Berechnung nach p(n) (für n=|w|) Schritten abbricht und sich sonst wie M verhält. Es gilt $t'_{M'} \leq p(n)$, da es auf keinem Wort der Länge n einen Pfad länger p(n) geben kann. Der kürzeste akzeptierende Pfad bleibt hierbei erhalten, da er nicht länger als p(n) sein kann. Also gilt $L(M') = L$ und deshalb L $\in$ NP'.

NP' $\subseteq$ NP:
Sei L $\in$ NP, d.h. es existiert eine NTM M' mit $L(M') = L$ und Polynom p mit $t'_{M'}(n) \leq p(n)$. Dann ist $t_{M'} \leq p(n)$, da alle Pfade polynomiell beschränkt sind, insbesondere auch der kürzeste akzeptierende. Daher L $\in$ NP.

NP' $\subseteq$ NP $\wedge$ NP $\subseteq$ NP' $\implies$ NP' = NP.