Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2021 – Hausaufgabenblatt 13

- Diese Woche werden alle Aufgaben korrigiert.
- Wenn Sie einen Beweis aufstellen, von dem Sie wissen, das einzelne Schritte problematisch oder unvollständig sind, merken Sie dies bitte in Ihrer Lösung an, damit wir das bei der Korrektur positiv berücksichtigen können.
- Um die Korrektheit einer Reduktion zu argumentieren, genügt es, die wesentlichen Argumente hinter Ihrer Konstruktion zu nennen. Ein vollständig präzises formales Argument ist nicht notwendig.
- $0 \in \mathbb{N}$

Aufgabe H13.1. (*2-EZ*) 0.5+0.5+0.5+0.5+0.5+0.5 Punkte

Bestimmen Sie für folgende Probleme jeweils, ob sie NP-vollständig sind, unter der Annahme $\mathsf{P} \neq \mathsf{NP}$. Begründen Sie Ihre Antworten *kurz*.

(a) **Eingabe:** Aussagenlogische Formel F
Ausgabe: Existieren zwei erfüllende Belegungen für F ?

(b) **Eingabe:** Wörter $u, v \in \{0,1\}^*$
Ausgabe: Gibt es ein Wort $w \in \{0,1\}^{|uv|}$, sodass auf Eingabe w die DTM M_u in weniger Schritten als die DTM M_v hält? (M_v muss nicht halten.)

(c) **Eingabe:** Zwei endliche Mengen $A, B \subset \mathbb{N}$, $k \in \mathbb{N}$
Ausgabe: Gibt es $M \subseteq B$, $|M| \leq k$, sodass jedes $x \in A$ ein Element in M teilt?

(d) **Eingabe:** Aussagenlogische Formel F
Ausgabe: Existiert eine *äquivalente* Formel F' in konjunktiver Normalform?

(e) **Eingabe:** Vektoren $v_1, ..., v_k \in \mathbb{Z}^d$, $d \in \mathbb{N}$
Ausgabe: Existieren $x_1, ..., x_n, y_1, ..., y_m \in \{v_1, ..., v_k\}$, sodass $\sum_i x_i - \sum_i y_i = 0$, $n + m > 0$, und $\{x_1, ..., x_n\} \cap \{y_1, ..., y_m\} = \emptyset$?

(f) **Eingabe:** Ungerichteter Graph $G = (V, E)$
Ausgabe: Gibt es einen Kreis in G, der jeden Knoten und jede Kante genau einmal besucht? (Start und Ende werden nicht separat gezählt.)

Lösungsskizze.

(a) Wahr. Für (a) $\in$ NP verwenden wir das Paar der Belegungen als Zertifikat. Außerdem können wir von SAT reduzieren, indem wir eine nutzlose Variable hinzufügen.

(b) Falsch. Das Problem ist unentscheidbar; wir reduzieren leicht von Halteproblem auf leerem Band, indem wir die TM so verändern, dass sie ihre Eingabe löscht, und als M_u verwenden. M_v ist dann eine TM, die nie terminiert.

(c) Wahr. Das Zertifikat ist M, und für die andere Richtung reduzieren wir von MENGENÜBERDECKUNG: Jedes Element wird zu einer unterschiedlichen Primzahl, und jede Menge zum Produkt der Primzahlen ihrer Elemente. Um n Primzahlen zu finden, gehen wir n^2 Zahlen durch und testen, ob die Zahl prim ist. Dass wir so immer genug Primzahlen finden, folgt z.B. aus dem Primzahlsatz.

(d) Falsch. Die Antwort ist immer „ja“, und das Problem somit in P. Wenn es NP-vollständig wäre, folgt $\mathsf{P} = \mathsf{NP}$ nach Lemma 6.16, entgegen der Annahme.

(e) Falsch. Das Problem ist in P, da es genügt zu bestimmen, ob die Vektoren $v_1, ..., v_k$ linear abhängig sind (mit Gauß-Elimination). Die Summe $\sum_i x_i - \sum_i y_i$ ist nach den Einschränkungen eine nichttriviale Linearkombination der v_i, falls diese also linear unabhängig sind, kann die Summe nie 0 sein. Umgekehrt gilt, dass, falls die v_i linear abhängig sind, nichttriviale Faktoren $\lambda_1, ..., \lambda_k \in \mathbb{Q}$ existieren, sodass $\sum_i \lambda_i v_i = 0$. Indem wir mit dem Produkt der Nenner multiplizieren, erhalten wir Faktoren $\lambda'_i \in \mathbb{Z}$. Wenn $\lambda'_i > 0$, dann wählen wir v_i entsprechend häufig als x_j, ansonsten als y_j.

(f) Falsch. Ein Kreis, der jede Kante besucht, besucht jeden Knoten mit Grad d genau $d/2$-mal. Wir müssen also lediglich überprüfen, ob der Graph zusammenhängend ist und alle Knoten Grad 2 haben.

Aufgabe H13.2. (*There are four colours! Four!*) 1+1+0.5+0.5+1+1 Punkte

Sei $G = (V, E)$ ein ungerichteter Graph. Für $k \in \mathbb{N}$ ist eine *k-Färbung* von G eine (totale) Funktion $f : V \to \{0, ..., k-1\}$ mit $f(u) \neq f(v)$ für alle $\{u, v\} \in E$. Es haben also alle benachbarten Knoten unterschiedliche Farben. Das *k-Färbbarkeitsproblem* (k-COL) ist folgendermaßen definiert, für $k \in \mathbb{N}$.

Eingabe: Ungerichteter Graph $G = (V, E)$
Ausgabe: Existiert eine k-Färbung für G ?

Unser Ziel ist es, zu zeigen, dass 3-COL ein NP-vollständiges Problem ist. Als Zwischenschritt betrachten wir eine Variante des Problems, bei dem eine Teilfärbung bereits vorgegeben ist. Für eine k-Färbung f ist eine *Teilfärbung* $f' : V \to \{0, ..., k-1\}$ eine beliebige partielle Funktion mit $f'(v) \in \{f(v), \perp\}$ für alle v. Das *partielle k-Färbbarkeitsproblem* (Partial-k-COL) ist nun

Eingabe: Ungerichteter Graph $G = (V, E)$, partielle Funktion $f' : V \to \{0, ..., k-1\}$
Ausgabe: Existiert eine k-Färbung f für G, mit Teilfärbung f' ?

(a) Zeigen Sie Partial-k-$\mathsf{COL} \leq_p k$-COL und k-$\mathsf{COL} \leq_p$ Partial-k-COL für $k \in \mathbb{N}$. Es genügt hier, die Korrektheit Ihrer Reduktionen zu skizzieren, ein vollständiger Beweis ist nicht notwendig.

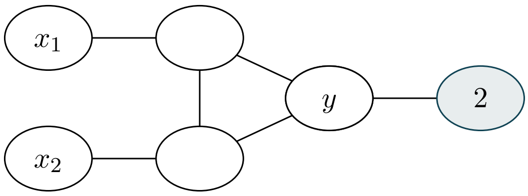
(b) Beweisen Sie k-$\mathsf{COL} \in \mathsf{NP}$ für $k \in \mathbb{N}$.

Nun wollen wir SAT auf Partial-3-COL reduzieren, indem wir die Farben verwenden, um logische Gatter zu simulieren. Wir verwenden Farben $\{0, 1, 2\}$, wobei 0 und 1 für wahr und falsch stehen, und 2 eine Farbe ist, die wir intern in unserem Schaltkreis verwenden, allerdings nicht als Eingabe oder Ausgabe. Zusätzlich betrachten wir mehrdeutige Gatter, die bei bestimmten Eingaben sowohl 1 als auch 0 ausgeben können.

Seien nun $n \in \mathbb{N}$ und $g : \{0, 1\}^n \to \mathcal{P}(\{0, 1\})$ beliebig. Wir wollen eine Instanz von Partial-k-COL konstruieren, also einen partiell gefärbten Graphen, die einen Schaltkreis darstellt, der g simuliert. Dazu verwenden wir Knoten $x_1, ..., x_n$ und y, wobei $x_1, ..., x_n$ die Eingabe abbilden (also mit beliebigen Farben in $\{0, 1\}$ gefärbt werden können), und y dann eine Farbe entsprechend g erhalten muss.

Formal sei $G = (V \cup \{x_1, ..., x_n, y\}, E)$ ein Graph, und $f' : V \to \{0, 1, 2\}$ eine partielle Funktion. Wir sagen, dass (G, f') ein *g-Gatter* ist, wenn für jede Funktion $I : \{x_1, ..., x_n\} \to \{0, 1\}$ eine 3-Färbung f von G existiert, sodass f' und I Teilfärbungen von f sind. Die Menge der Farben, die solche f dem Knoten y zuordnen können, muss genau $g(I(x_1), ..., I(x_n))$ entsprechen, es gilt also $\{f(y) : f\} = g(I(x_1), ..., I(x_n))$.

Die folgende Instanz ist z.B. ein `CHOICE`-Gatter, wobei $\texttt{CHOICE}(x_1, x_2) := \{x_1, x_2\}$ für $x_1, x_2 \in \{0, 1\}$. (Die Ausgabe entspricht also einer der Eingaben, darf sich aber aussuchen, welcher.)



(c) Konstruieren Sie ein `NOT`-Gatter, wobei $\texttt{NOT}(x_1) := \{\neg x_1\}$ für $x_1 \in \{0, 1\}$.

(d) Sei $\texttt{weakNOT}(x_1) := \texttt{NOT}(x_1) \cup \{0\}$. Konstruieren Sie ein `weakNOT`-Gatter.

(e) Konstruieren Sie ein `OR`-Gatter, mit $\texttt{OR}(x_1, x_2) := \{x_1 \vee x_2\}$ für $x_1, x_2 \in \{0, 1\}$.

(f) Beweisen Sie nun SAT $\leq_p$ Partial-3-COL.

Hinweis: Wenn Sie ein Objekt konstruieren, und es nicht offensichtlich ist, dass ihre Konstruktion den geforderten Bedingungen entspricht, begründen Sie dies kurz.

Lösungsskizze. **(a)** k-COL $\leq_p$ Partial-k-COL ist trivial, wir verwenden einfach die partielle Funktion, die nirgendwo definiert ist und lassen den Graph unverändert.

Für die andere Richtung sei nun $G = (V, E)$ ein Graph und $f' : V \to \{0, ..., k{-}1\}$ eine partielle Funktion. Wir fügen G eine k-Clique hinzu, mit Knoten $v_0, ..., v_{k-1}$. Wir wissen, dass jede Färbung von G den Knoten $v_0, ..., v_{k-1}$ unterschiedliche Farben zuweisen muss (wir ignorieren f' zunächst). Bei einer Färbung Farben zu tauschen macht die Färbung nicht ungültig, wir können also oBdA annehmen, dass v_i genau Farbe i bekommt, für $i \in \{0, ..., k{-}1\}$. Damit können wir nun die partielle Färbung realisieren: hier verbinden wir jeden Knoten $u \in V$, für den $f'(u) \neq \perp$ gilt, mit den Knoten $\{v_i : i \neq f'(u)\}$. Damit muss u dann zwangsläufig Farbe $f'(u)$ haben.

Die Aufgabenstellung fordert nur eine Beweisskizze der Korrektheit und die bisherigen Ausführungen genügen dafür. Zwecks Verständlichkeit folgt nun ein ausführlicher Beweis.

Formal definieren wir also $G' := (V', E')$ mit $V' := V \uplus \{v_0, ..., v_{k-1}\}$ und

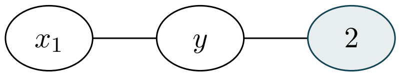
$$E' := E \cup \{v_i, v_j : i \neq j\} \cup \{\{u, v_i\} : u \in V, i \neq f(u) \neq \perp\}$$

Wenn eine Färbung f für G' existiert, dann existiert auch eine Färbung f^* mit $f^*(v_i) = i$ für $i \in \{0, ..., k{-}1\}$. (Dies gilt, da wir Farben vertauschen können, ohne die Färbung zu beeinflussen.) Angenommen, f' wäre keine Teilfärbung von f^*. Dann gibt es ein $u \in V$ mit $\perp \neq f'(u) \neq f^*(u)$. Sei nun $c := f^*(u)$. Da $\perp \neq f'(u)$ und $f'(u) \neq c$, gibt es eine Kante $\{u, v_c\}$. Aber $f^*(v_c) = c = f^*(u)$; dies ist ein Widerspruch dazu, dass f^* eine Färbung ist.

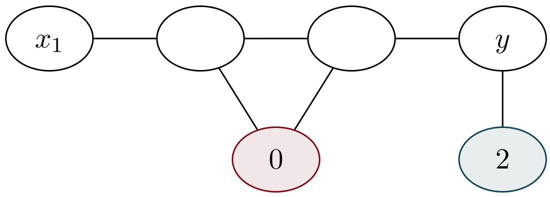
Umgekehrt, wenn $f : V \to \{0, ..., k{-}1\}$ eine Färbung von G mit Teilfärbung f' ist, können wir f auf V' erweitern, indem wir $f(v_i) := i$ definieren, für $i \in \{0, ..., k{-}1\}$. Jede Kante in E hat unterschiedliche Farben, da f eine Färbung ist. Jede Kante in $\{v_0, ..., v_{k-1}\}$ auch, da $f(v_i) = i$. Und schließlich sind auch die Kanten zwischen einem $u \in V$ und einem v_i erfüllt, da diese Kanten nur existieren, wenn $f'(u)$ definiert ist; somit gilt $f(u) = f'(u) \neq i = f(v_i)$.

(b) Hierzu verwenden wir die Färbung als Zertifikat. Wir verifizieren die Färbung, indem wir alle Kanten durchgehen, was polynomielle Zeit benötigt. Die Färbung besteht aus $|V|$ Zahlen, ist also auch nur polynomiell lang.

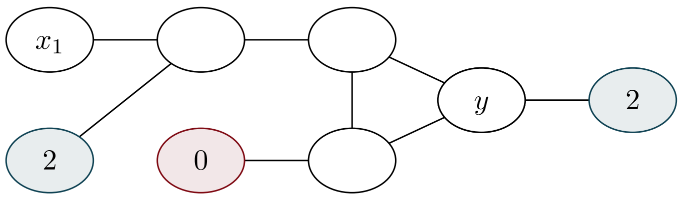
(c)



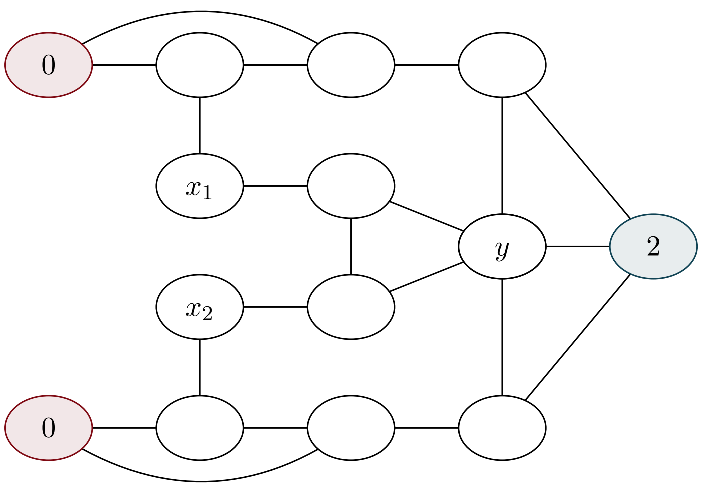
(d) Hier geben wir zwei mögliche Lösungen an. Zunächst kann man das Gatter folgendermaßen auf direktem Wege bauen:



Wenn $x_1 = 1$, dann muss der nächste Knoten 2 sein, der nächste 1 und y schließlich 0. Für $x_1 = 0$ können wir den nächsten Knoten als 1 wählen, den folgenden als 2 und y dann als 0 oder 1. Alternativ lässt es sich auch über das `CHOICE`-Gatter konstruieren, da $\texttt{weakNOT}(x_1) = \texttt{CHOICE}(\texttt{NOT}(x_1), 0)$, also als



(e) Wenn wir nur $\texttt{CHOICE}(x_1, x_2)$ verwenden, sind wir für die Fälle $x_1 = x_2$ bereits fertig. Für die Fälle $x_1 \neq x_2$ müssen wir dann allerdings dafür sorgen, dass die Ausgabe 1 und nicht 0 ist. Hierfür können wir zwei `weakNot`-Gatter verwenden: Für x_1 und x_2 bauen wir jeweils ein solches Gatter und verbinden dessen Ausgabe mit der Ausgabe des `CHOICE`-Gatters. Wenn z.B. $x_1 = 0$, dann kann das `weakNot`-Gatter einen beliebigen Wert annehmen und beeinflusst die Ausgabe des `CHOICE`-Gatters nicht. Wenn allerdings $x_1 = 1$, dann ist hat das `weakNot`-Gatter Ausgabe 0, und das `CHOICE`-Gatter muss somit Ausgabe 1 haben.



(f) Mit den Gattern für `NOT` und `OR` können wir auch ein Gatter für `AND` bauen, und damit beliebige Schaltkreise. Wir reduzieren nun von SAT, sei also F eine beliebige aussagenlogische Formel mit Variablen $x_1, ..., x_k$. Wir definieren uns nun die Funktion $g : \{0,1\}^k \to \mathcal{P}(\{0,1\})$, sodass für eine beliebige Belegung $\sigma : \{x_1, ..., x_k\} \to \{0,1\}$ gilt, dass $g(\sigma(x_1), ..., \sigma(x_k)) := \{\sigma(F)\}$. Die Funktion g bildet also die Werte der Variablen auf die einelementige Menge, die den Wahrheitswert der Formel beinhaltet, ab. Nun können wir ein g-Gatter bauen, unter Verwendung der `NOT`-, `OR`- und `AND`-Gatter. Schließlich müssen wir sicherstellen, dass eine Färbung den Knoten $x_1, ..., x_k$ eine erfüllende Belegung zuweist. Hierzu erstellen wir einen Knoten, den wir mit Farbe 2 färben, und verbinden ihn mit jedem x_i, für $i \in \{1, ..., k\}$, damit diese nur noch Farben 0 und 1 erhalten können. Um dafür zu sorgen, dass die Belegung erfüllend ist, weisen wir Knoten y die Farbe 1 zu.

Aufgabe H13.3. (*Vollständig, oder fehlt da was?*) 1+1 Punkte

Zeigen Sie, dass das folgende Problem, was wir NTM-SIM nennen, NP-vollständig ist, indem Sie Definitionen 6.14 und 6.15 direkt verwenden. Sie dürfen also insbesondere nicht benutzen, dass wir von anderen Problemen (z.B. SAT) bereits gezeigt haben, dass diese NP-schwierig sind.

Eingabe: $v\$x\$1^n$ für $v, x \in \{0,1\}^*$, $n \in \mathbb{N}$
Ausgabe: Akzeptiert die NTM M_v die Eingabe x in höchstens n Schritten?

(a) Zeigen Sie NTM-SIM $\in$ NP .

(b) Beweisen Sie, dass NTM-SIM NP-schwierig ist.

Sie dürfen ohne Angabe von Details verwenden, dass eine Turingmaschine M eine beliebige andere TM M_w simulieren kann, auch wenn sie nur über w verfügt.

Lösungsskizze. **(a)** Wir können dieses Problem leicht mit einer NTM lösen. Hierfür simulieren wir M_v auf der Eingabe x für n Schritte – wenn M_v akzeptiert, tun wir dies auch. Da eine Eingabe $w = v\$x\$1^n$ Länge mindestens $|w| \geq n$ hat, benötigen wir nur polynomiell viel Zeit hierfür. Somit kann das Problem in Polynomialzeit auf einer NTM entschieden werden, und ist in NP.
(b) Sei $P \in$ NP beliebig, und sei M_v eine NTM, die P in polynomieller Zeit entscheidet. Insbesondere gibt es also ein Polynom p, sodass M_v jedes $x \in P$ in höchstens $p(|x|)$ Schritten akzeptiert. Unsere Reduktionsfunktion f ist nun definiert als $f(x) := v\$x\$1^{p(|x|)}$. Dies lässt sich offensichtlich in polynomieller Zeit berechnen, und die Länge der Eingabe ist polynomiell in $|x|$.

Aufgabe H13.4. (~~*Auf Wiedersehen!*~~ *Tschüss!*) 1+1 Punkte

Theo hat bald die Grundschule erfolgreich abgeschlossen, und wechselt auf eine weiterführende Schule. Dora freut sich riesig für ihn, und möchte eine große Abschiedsparty organisieren. Sie hat auch schon eine lange Liste mit Freunden und Bekannten, die sie einladen könnte. Leider gibt es ein paar Personen, die sich nicht leiden können und deswegen nicht gleichzeitig eingeladen werden sollten. Dora versucht herauszufinden, wie viele Leute sie maximal einladen kann. Können Sie ihr dabei helfen?

Wir betrachten hier das PARTY Problem.
Eingabe: Endliche Menge F an Freunden, Abneigungen $A \subseteq F \times F$, $k \in \mathbb{N}$
Ausgabe: Existiert ein $M \subseteq F$ mit $|M| \geq k$ und $M^2 \cap A = \emptyset$?

(a) Dora hat folgende Liste:

1. Theo
2. Dora
3. Bijanka
4. Theos Neffe
5. Eva Pirsalz
6. Jasper Vazarie
7. Tyx'Hxli E'Oozrt
8. Ehemaliger Rektor
9. Dokh'Toa Ivl'Zpaasa
10. Dr. Evilsparza
11. Kindergartenprofessor
12. Baukommissionsleiter

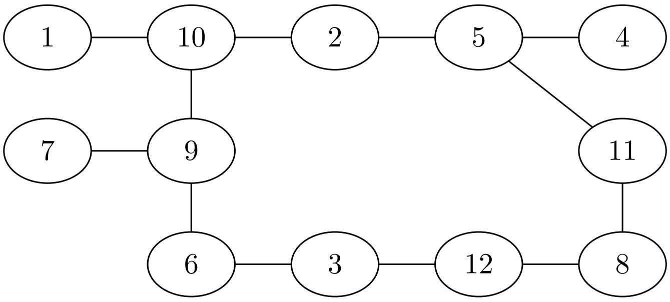
Folgende Personen verstehen sich nicht so gut und dürfen deshalb nicht beide eingeladen werden.

$$\{(2,5),(7,9),(8,12),(1,10),(5,11),(3,6),\\ (4,5),(3,12),(9,10),(6,9),(2,10),(8,11)\}$$

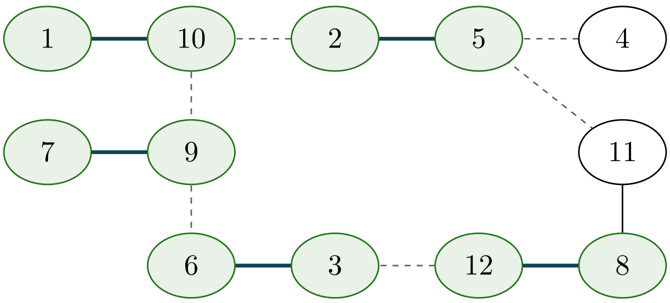
Wie viele Personen kann Dora maximal einladen? Beweisen Sie Ihre Antwort, und somit insbesondere, wieso es unmöglich ist, mehr Personen einzuladen.

(b) Zeigen Sie, dass PARTY ein NP-vollständiges Problem ist.

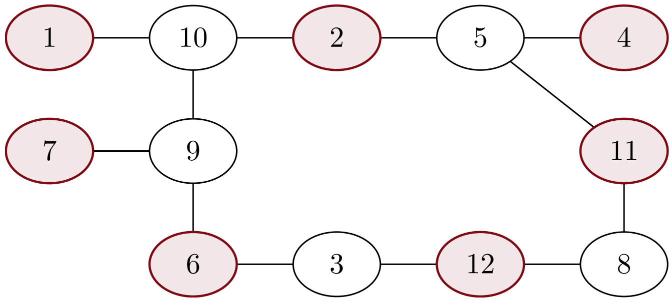
Lösungsskizze. **(a)** Wir können diese Liste als Graphen betrachten:



Dies ist genau der ungerichtete Graph $G = (F, A)$. In diesem Graphen finden wir nun ein Matching, also eine Teilmenge der Kanten, sodass keine zwei Kanten adjazent sind:



Von jeder dieser Kanten können wir einen der adjazenten Knoten einladen. Wir haben 5 solcher Paare gefunden und $|F| = 12$, es ist also nicht möglich, mehr als 7 Personen einzuladen. Nun genügt es also, eine Lösung $M \subseteq F$ mit Größe $|M| = 7$ zu finden. Notwendigerweise müsste hierfür nach unserem vorherigen Argument $4, 11 \in M$ gelten (falls ein M mit Größe 7 existiert). Für jedes Paar müsste genau einer der Knoten eingeladen werden, also dann $2, 12$, danach $1, 6$, und schließlich 7. Wir erhalten $M = \{1, 2, 4, 6, 7, 11, 12\}$, was tatsächlich eine Lösung ist.



(b) Wie in (a) betrachten wir für das PARTY Problem den ungerichteten Graphen (F, A).

Sei f die Funktion, die auf Eingabe eines ungerichteten Graphen $G = (V, E)$ den komplementären Graphen $\overline{G} = (V, V^2 \setminus E)$ ausgibt, also den Graphen, in dem zwei Knoten genau dann adjazent sind, wenn sie es in G nicht sind. (Formal eigentlich $\overline{G} = (V, \overline{E})$ mit $\overline{E} := \{\{u, v\} : u, v \in V, u \neq v, \{u, v\} \notin E\})$, da ungerichtete Kanten keine Paare sind.) Offensichtlich lässt sich f in Polynomialzeit berechnen, da $|E| \leq |V|^2$ immer gelten muss.

Wir können nun PARTY $\leq_p$ CLIQUE und CLIQUE $\leq_p$ PARTY reduzieren. In beiden Fällen ist f die Reduktionsfunktion (den Parameter k lassen wir unverändert). Sei nun (F, A, k) eine Instanz von PARTY, und $G := (F, A)$. Eine Menge $M \subseteq F$ mit $|M| \geq k$ ist genau dann eine Lösung von (F, A, k), wenn keine zwei Knoten in M benachbart sind. Dies ist äquivalent dazu, dass in $f(G)$ alle Knoten in M benachbart sind, also M eine Clique in $f(G)$ bildet. Somit gilt

$$(G, k) \in \textsf{PARTY} \Leftrightarrow (f(G), k) \in \textsf{CLIQUE}$$

und wir haben PARTY $\leq_p$ CLIQUE gezeigt. Da f bijektiv ist, gilt ebenfalls

$$(G, k) \in \textsf{CLIQUE} \Leftrightarrow (f^{-1}(G), k) \in \textsf{PARTY}$$

Aufgrund von $f^{-1} = f$ folgt nun CLIQUE $\leq_p$ PARTY.

Da CLIQUE (wie in der Vorlesung gezeigt) NP-vollständig ist, ist es auch PARTY.

Wir hoffen, Sie hatten zumindest ein wenig Freude beim Bearbeiten der Aufgaben. Die Hausaufgabenzeit ist nunmehr bald vorbei – wir verabschieden uns und wünschen Ihnen fröhliche vorlesungsfreie Wochen!

—Ihr fiktionales THEO-Team

Theo & Dora