ETH Zürich
Institute of Theoretical Computer Science
Prof. Rasmus Kyng
Prof. Angelika Steger

FS 2024

Algorithmen und Wahrscheinlichkeit
Theorie-Aufgaben 3 — Lösung

Aufgabe 1 – *Jass-Karten*

(a) Der wohl leichteste und effizienteste Algorithmus besteht daraus, 'Ja' auszugeben und den Algorithmus zu beenden. Die Korrektheit folgt aus Aufgabenteil (b).

(b) Wir modellieren das Problem graphentheoretisch: Wir konstruieren einen bipartiten Graphen G, mit Knotenklassen W und S. Die Klasse W besteht aus 9 Knoten, namentlich den neun Werten 6, 7, 8, 9, 10, Under, Ober, König, Ass. Die Klasse S besteht ebenfalls aus neun Knoten, nämlich den neun Stapeln $S_1, \ldots, S_9$. Für $W_i \in W$ und $S_i \in S$ fügen wir die Kante $\{W_i, S_i\}$ genau dann ein, wenn der Wert W_i im Stapel S_i vorkommt. Der so konstruierte Graph G ist offensichtlich bipartit.

Es ist nun leicht einzusehen, dass eine Kartenauswahl, die aus jedem Stapel eine Karte enthält und eine vollständigen Strasse ergibt, genau einem perfekten Matching zwischen W und S entspricht. Es bleibt also zu beweisen, dass G ein perfektes Matching enthält und dieses algorithmisch effizient zu finden.

Wir beweisen zunächst die Existenz eines perfekten Matchings. Da unser Graph G bipartit ist, genügt es zu zeigen, dass G Halls Bedingung erfüllt - also dass für alle $X \subseteq W$ gilt, dass $|N_G(X)| \geq |X|$. Betrachten wir also eine Menge $X \subseteq W$ von Werten. Nach Definition des Graphen G müssen für jedes $W_i \in X$ alle vier Karten vom Wert W_i in den Stapeln aus $N_G(X)$ enthalten sein. Die Stapel aus $N_G(X)$ müssen zusammen also mindestens $4|X|$ Karten enthalten. Andererseits enthalten sie genau $4|N_G(X)|$ Karten, da ja jeder Stapel genau 4 Karten enthält. Also ist $4|N_G(X)| \geq 4|X|$, woraus wie gewünscht $|N_G(X)| \geq |X|$ folgt. Damit ist Halls Bedingung erfüllt, und G enthält nach dem Satz von Hall ein perfektes Matching. Ausserdem können wir dieses mit dem in der Vorlesung diskutierten Algorithmus in Zeit $O(n \cdot m)$ finden, bzw. sogar in Zeit $O(\sqrt{n} \cdot m)$. Das gefundene Matching lässt sich leicht in eine vollständige Strasse überführen.

Ein alternativer Beweis könnte statt G auch einen Multigraphen konstruieren, der Kanten mehrfach enthält, wenn ein Wert mehrfach auf einem Stapel vorkommt. Der so entstehende Multigraph ist 4-regulär, wir können also im Prinzip den Satz von König (Satz 1.49 im Skript) anwenden. Dazu müssten wir jedoch erst zeigen, dass der Satz von König auch für Multigraphen gilt, was ähnlich aufwendig ist wie die direkte Anwendung des Satzes von Hall. Tatsächlich funktioniert der in der Vorlesung angegebene Algorithmus (Satz 1.49) auch in Multigraphen, sodass wir auf diese Weise sogar einen Laufzeit $O(m)$ erreichen, wobei m hier die Zahl der Spielkarten ist (vorausgesetzt, dass die Zahl der Werte eine Zweierpotenz ist).