



10. Übungsblatt zu Grundzüge von Algorithmen und Datenstrukturen, WS 15/16

Prof. Markus Bläser

<http://www-cc.cs.uni-saarland.de/course/50/>

Abgabe: 21. Januar 2016, 12:00 Uhr

Aufgabe 10.1 Am Ende des Semesters schreiben Sie n Klausuren. Zur Vorbereitung stehen Ihnen h Stunden zur Verfügung. Zu jeder Klausur i gibt es eine Funktion $f_i : \{0, \dots, h\} \rightarrow \{0, \dots, p\}$. Wenn Sie j Stunden in die Vorbereitung für Klausur i verwenden, so erhalten Sie $f_i(j)$ Punkte in der Klausur. Die Funktionen f_i sind monoton wachsend. Schreiben Sie einen Algorithmus, der berechnet, wie viele Stunden Sie auf die einzelnen Klausuren jeweils verwenden sollen, um die durchschnittliche Punktzahl in den Klausuren zu maximieren. Die Laufzeit soll polynomiell in n , p und h sein. Die Werte $f_i(j)$ können zu Einheitskosten berechnet werden.

- Aufgabe 10.2**
- Sei $G = (V, E)$ ein gerichteter Graph und $s \in V$. Ein Knoten $t \in V$ heißt von s erreichbar, wenn es einen gerichteten Pfad von s nach t gibt. Wie muss man die Breitensuche modifizieren, so dass man in einem gerichteten Graph alle von s erreichbaren Knoten findet?
 - Ein Teilmenge $C \subseteq V$ heisst *starke Zusammenhangskomponente*, wenn für alle $u, v \in C$ gilt, dass u von v erreichbar ist und v von u erreichbar ist, und es kein $C' \supsetneq C$ gibt, dass diese Eigenschaft besitzt. Zeigen Sie, dass es genau eine starke Zusammenhangskomponente gibt, zu der ein Knoten s gehört.
 - Geben Sie einen Algorithmus an, der die starke Zusammenhangskomponente berechnet, zu der ein gegebener Knoten s gehört.

Aufgabe 10.3 Gegeben sei ein ungerichteter Graph $G = (V, E)$ und $s_1, s_2, t_1, t_2 \in V$ paarweise verschiedene Knoten. Der Abstand zwischen zwei Knoten u und v ist die Länge eines kürzesten Pfades zwischen u und v . Wir nehmen an, dass s_1 und s_2 mindestens Abstand r haben für ein gegebenes $r \in \mathbb{N}$ und ebenso t_1 und t_2 . Stellen sie sich vor, zwei Roboter befinden sich in den Punkten s_1 und s_2 . In jedem Zeitschritt kann sich ein Roboter entweder zu einem benachbarten Knoten bewegen oder stehen bleiben. Dabei soll der Abstand zwischen den beiden Robotern immer mindestens r betragen. Entwerfen Sie einen Algorithmus, der ein Schema berechnet, dass mit möglichst wenig Zeitschritten den einen Roboter nach t_1 und den anderen nach t_2 bewegt und dabei die obige Abstandsregel einhält. Analysieren Sie dessen Laufzeit. Können Sie Ihren Algorithmus auf eine beliebige Anzahl von Robotern erweitern?