Technische Universität München
Institut für Informatik
Prof. Dr. Angelika Steger

SS 2003
Merkblatt 1
07.04.2003

Praktikum Diskrete Optimierung

Durchführung des Praktikums

Hintergrund

Unter Diskreter Optimierung versteht man die Suche nach einer optimalen Lösung zu einem Problem mit diskretem Suchraum. Solche Probleme treten in der Praxis an vielen Stellen auf. Als Beispiele seien genannt:

- Einplanung von Aufträgen bei Fertigungsstraßen
- Zuteilung von Verbindungen in Kommunikationsnetzwerken
- Berechnung des Chiplayouts bei VLSI

Viele Diskrete Optimierungsprobleme, die in der Praxis auftreten, lassen sich sehr gut mit Hilfe von Graphen modellieren und lösen. Aus diesem Grund beschäftigen sich seit mehreren Jahrzehnten Wissenschaftler aus den Bereichen Informatik und Mathematik intensiv mit der Entwicklung von Algorithmen zur effizienten Lösung typischer Problemstellungen für Graphen. Doch was ist ein effizienter Algorithmus? Gewöhnlich lassen sich die bei der Diskreten Optimierung betrachteten Probleme dadurch charakterisieren, daß der Suchraum endlich ist und somit eine optimale Lösung durch vollständige Suche gefunden werden könnte. Da der Suchraum jedoch meistens zwar endlich aber riesig ist (und die exakte Lösung der Probleme nicht selten NP-schwer ist), muß man in der Praxis auf andere Methoden ausweichen. Hierbei gilt es, die Struktur der zugrundeliegenden Probleme bzw. Graphen bestmöglich auszunutzen. Tatsächlich hat sich gezeigt, daß es für viele Probleme, bei denen ein naiver Lösungsansatz zu exponentiellen Laufzeiten führt, doch sehr schnelle Algorithmen gibt.

Inhalt des Praktikums

Im Rahmen des Praktikums sollen Student(inn)en zu einem großen Teil verschiedene Graphenalgorithmen effizient implementieren. Während Probleme mit kleinen Eingabegrößen in der Regel auch mit naiven ad hoc Ansätzen gelöst werden können, zeigen sich die Grenzen derartiger Verfahren bei steigender Problemgröße und besonders bei praxisrelevanten Eingabedaten recht schnell. Der Fokus im Praktikum Diskrete Optimierung liegt daher auf der Lösung von Problemen mit großen oder sogar riesigen Eingabeinstanzen. Wir werden im Praktikum den Schwerpunkt auf Matching-Algorithmen legen.
Mit dem Praktikum werden mehrere Ziele verfolgt:

- Tieferes Verständnis der Methoden zur Diskreten Optimierung,

- Erfahrungen bei der Entwicklung hochoptimierter Software für rechenintensive Probleme.
- Spaß an der Herausforderung, die bestmögliche Leistung aus dem verfügbaren Rechensystem "herauszukitzeln".

Vorkenntnisse aus den Veranstaltungen Effiziente Algorithmen und Datenstrukturen und Effiziente Algorithmen und Datenstrukturen II sind nützlich, können aber auch parallel zum Praktikum erworben werden.

Aufgaben des Praktikums

Im Laufe des Semesters werden voraussichtlich 7 Aufgabenblätter mit Programmieraufgaben zu bearbeiten sein, die unter
`http://www14.in.tum.de/lehre/2003SS/optprak/data.html`
heruntergeladen werden können bzw. bei den Besprechungsterminen verteilt werden.
Für die Bearbeitung der Aufgabenblätter werden jeweils eine oder mehrere Wochen zur Verfügung stehen. Die Lösungen sollen in C++ zunächst ohne Verwendung zusätzlicher Bibliotheken programmiert werden.
Die Bearbeitung der Praktikumsaufgaben kann entweder auf unseren Lehrstuhl-Rechern (MI 03.09.034), in der Sun-Halle oder auch zu Hause erfolgen. In jedem Fall müssen sich die abgegebenen Programme mit einem g++, Version ≥ 2.95 kompilieren lassen.

Scheinerwerb und Abgabe

Die Aufgaben sollen in Zweierteams bearbeitet werden, wobei wir dringend empfehlen, die Aufgaben nicht aufzuteilen, sondern wirklich in Zusammenarbeit zu lösen und zu implementieren. Am Ende des Praktikums findet eine mündliche Prüfung statt, bei der erwartet wird, daß jede(r) Student(in) bei allen Praktikumsaufgaben auch zur Implementierung, die von seiner/ihrer Gruppe abgegeben wurde, Fragen beantworten kann.
Zum Testen der Programme stehen unter
`http://www14.in.tum.de/lehre/2003SS/optprak/data.html`
einige (auch große) Testeingaben zur Verfügung.
Wenn ein Team der Meinung ist, eine Aufgabe korrekt gelöst zu haben, soll das Programm *inklusive eines passenden Makefiles* per E-Mail an

`optprak@in.tum.de`

abgegeben werden. Diese Lösungen werden durchgesehen, sowohl auf den zur Verfügung gestellten Testeingaben als auch auf weiteren Beispieldaten getestet und auf effiziente Implementierung geprüft. Abgegebene Lösungen werden folgendermaßen bewertet:

- **ACCEPTED** Programm korrekt.

- **WRONG ANSWER** Programm liefert auf einer (oder mehreren) Testeingaben falsche Ergebnisse. Die jeweiligen Testeingaben oder zusätzliche Informationen zum Fehler werden *nicht* zur Verfügung gestellt.

- **TIME LIMIT EXCEEDED** Programm benötigt mehr als die vorgesehene Laufzeit.

- **RUNTIME ERROR** Programm stürzt bei einer oder mehreren Testeingaben ab.

- **COMPILE TIME ERROR** Programm kompiliert nicht.

Bis zum jeweils auf den Aufgabenblättern vermerkten Abgabetermin können dann korrigierte Versionen nachgereicht werden.
Eine Lösung wird also **nur dann akzeptiert**, falls das Programm

- *korrekt und effizient* ist,

- *rechtzeitig* abgegeben wurde,

- ausreichend *kommentiert* ist und

- *nicht abgeschrieben* wurde.

Teammitglieder erhalten einen Schein, wenn

- alle Aufgaben bearbeitet werden,

- **alle bis auf höchstens eine** abgegebene Lösung des Teams akzeptiert werden *und*

- die mündliche Prüfung bestanden wird.

Termine der Praktikumsbesprechung und Sprechstunden

Jeden Montag zwischen $14.^{15}$ und $15.^{45}$ Uhr gibt es eine (optionale) Vorbesprechung, in der die neuen Aufgaben erläutert werden.
Daneben werden pro Woche zwei Sprechstunden angeboten, in denen Fragen zu den Praktikumsaufgaben beantwortet werden und in denen auch die E-Mails mit Fragen zu den Praktikumsaufgaben bearbeitet werden. Die Sprechstunden finden jeweils Dienstags von $14.^{00}$ bis $16.^{00}$ Uhr und Freitags von $10.^{00}$ bis $12.^{00}$ Uhr statt.