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

SS 1999
Aufgabenblatt 2
18. Mai 1999

Praktikum Diskrete Optimierung

(Abgabetermin: Dienstag, den 25. Mai 1999, 12.00 Uhr)

Aufgabe 1 Komplexität Branch-and-Bound

Wieviele Lösungen werden bei Branch-and-Bound betrachtet, wenn man annimmt, daß die Bound-Bedingung nie greift. Der Graph habe n Knoten und die Pfade sollen maximal 2 Hops enthalten. Eine größenordungsmäßige Angabe genügt.

Aufgabe 2 Implementierung Branch-and-Bound

Schreiben Sie ein Programm

```
branchAndBound <max. time in min> <basename>
```

das die optimale Lösung mit der Branch-and-Bound-Technik ermittelt.
Diese Aufgabe läßt sich in folgende Teilschritte zerlegen:

- Erweitern Sie Ihre Datenstruktur zur Repräsentation einer Lösung des Problems. Die erweiterte Fassung sollte es ermöglichen, eine Lösung Schritt für Schritt durch Hinzufügen von Pfaden zu erzeugen bzw. durch Löschen von Pfaden wieder abzubauen.

  Die Kosten, die durch eine solche unvollständige Lösung verursacht werden (d. h. nur durch die Netzverbindung, die für die bislang festgelegten Pfade benötigt werden), sollten inkrementell berechnet werden.

- Überlegen Sie sich eine Heuristik zur Abschätzung der Kosten, die durch eine unvollständige Lösung mindestens verursacht werden, wenn sie optimal zu einer vollständigen Lösung erweitert wird. Diese Kosten sollen *unterschätzt* werden. Erläutern Sie die Idee Ihrer Heuristik durch ausführliche Kommentare im Sourcecode.

- Implementieren Sie den eigentlichen Branch-and-Bound-Algorithmus. Die maximale Anzahl von Hops sei zwei. Kann das Programm die optimale Lösung für die Datei „net7.def“ in unter 10 Minuten finden?

Hinweise

Dieses Aufgabenblatt ist bewußt ein wenig kürzer gehalten, damit Sie Zeit haben, eventuellen Nachholbedarf von Blatt 1 zu erledigen.
Auch die Datenstrukturen dieses Blattes zur Verwaltung unvollständiger Lösungen werden Sie im Laufe des Praktikum mehrmals wiederverwenden.