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

SS 2003
Aufgabenblatt 6
16.06.2003

Praktikum Diskrete Optimierung

Abgabetermin: Montag, den 23.06.2003, 14.^{00} Uhr

String Matching

Gegeben sei ein Text T der Länge n (d.h. bestehend aus n Zeichen) und ein Pattern P der Länge m. Die Suche nach *allen* Vorkommen des Patterns P im Text T kann unter anderem mit Hilfe folgender Algorithmen geschehen:

- **Naiver Ansatz:** Probiere das Pattern an jeder Stelle aus
  Laufzeit: $O(n \cdot m)$
- **Knuth-Morris-Pratt:** Geschickte Verwendung endlicher Automaten
  Laufzeit: $O(n + m)$
- **Boyer-Moore:** Verwendung zweier Heuristiken
  Laufzeit: $O\left((n - m + 1) \cdot m + |\Sigma|\right)$, wobei $|\Sigma|$ die Größe des Eingabealphabets ist.

Aufgabe 7 Algorithmus von Knuth, Morris und Pratt

Implementieren Sie den Algorithmus von Knuth, Morris und Pratt zur Suche aller Vorkommen des Patterns P im Text T. Ihr Algorithmus soll eine Laufzeit von $O(n + m)$ besitzen. Als Eingabealphabet gilt der normale ASCII-Zeichensatz. Ihr C++-Programm soll das Pattern und den Text von der *Standardeingabe* lesen und die Positionen aller Vorkommen des Patterns im Text auf der *Standardausgabe* ausgeben.

Wichtiger Hinweis: Halten Sie das im folgenden angegebene Eingabe- und Ausgabeformat bitte genau (keinerlei zusätzliche Leerzeichen, Zeilenumbrüche etc.) ein! Die abgegebenen Programme werden u.a. auch mit Hilfe eines automatisierten Verfahrens auf verschiedenen Eingaben getestet.

Eingabeformat

Als Eingabe erhält Ihr Programm zunächst eine (möglicherweise sehr lange) Zeile mit dem Pattern P. Die folgenden Zeilen enthalten den Text T.

Beispieleingabe

```
bcd
bcdaaaaaaaaaaaaaaaaaaaaaaaabdcbcd
bcdklbcdllllllkkkkkkkmhgggjklbcd
```

Ausgabeformat

Ihr Programm soll *alle* Vorkommen des Patterns P im Text T ausgeben, jeweils eines pro Zeile. Hierzu werden die Zeichen des Eingabetextes der Reihe nach **beginnend von 0** durchnummieriert, Zeilenumbrüche zählen als **ein einzelnes** Zeichen mit. Für jedes Vorkommen von P in T soll dann jeweils die Postion des ersten übereinstimmenden Zeichens von P und T ausgegeben werden.

Beispielausgabe

```
0
30
34
39
63
```

Weitere Beispieleingaben und Beispielausgaben stehen unter
`http://www14.in.tum.de/lehre/2003SS/optprak/data.html`
zur Verfügung.

Aufgabe 8 Algorithmus von Boyer und Moore

Implementieren Sie den Algorithmus von Boyer und Moore zur Suche aller Vorkommen des Patterns P im Text T. Ihr Algorithmus soll eine Laufzeit von $O\left((n-m+1)\cdot m+|\Sigma|\right)$ besitzen. Als Eingabealphabet gilt auch hier der normale ASCII-Zeichensatz.
Ihr C++-Programm soll das Pattern und den Text von der *Standardeingabe* lesen und die Positionen aller Vorkommen des Patterns im Text auf der *Standardausgabe* ausgeben. Eingabe- und Ausgabeformat sind mit dem zu Aufgabe 6 beschriebenen Format identisch.

Zeitlimit und Abgabe

Das Zeitlimit sowohl für Aufgabe 7 als auch für Aufgabe 8 beträgt pro Testeingabe jeweils 60 Sekunden.
Die Abgabe der Lösungen **als einzelne .tar.gz-Files für jede Aufgabe** muß bis Montag, 23.06.2003, $14.^{00}$ Uhr, per E-Mail an die Adresse `optprak@in.tum.de` erfolgen.

Literaturhinweise

- T.H. Cormen, C.E. Leiserson, R.L. Rivest. Introduction to Algorithms. MIT Press.

Abschlussprüfungen

Die Termine der Abschlussprüfungen sind festgelegt und können auf der Status-Webseite unter `http://www14.in.tum.de/lehre/2003SS/optprak/status.html` abgefragt werden. Die Prüfungen werden pro Team ca. 10 min dauern. Die für die Praktikumsaufgaben abgegebenen und akzeptierten Lösungen müssen von allen Teammitgliedern zum Prüfungszeitpunkt im Praktikumsraum MI 03.09.034 schnell abrufbar sein. **Alle** Teammitglieder müssen ebenso in der Lage sein, über die Aufgabenstellungen und Lösungen (Idee und Komplexität der Algorithmen, Lösungswege) **aller** Praktikumsaufgaben Auskunft zu geben.