Technische Universität München
Institut für Informatik
Lehrstuhl für effiziente Algorithmen
Prof. Dr. E.W. Mayr
Prof. Dr. A. Steger

Proseminar
Textalgorithmen
Sommersemester 1998
18. Februar 1998

# Proseminar

# **Textalgorithmen**

Sommersemester 1998
Prof. Dr. E.W. Mayr
Prof. Dr. A. Steger

**Vorbesprechung:** Donnerstag, den 19. 2. '98, 13.30 Uhr, Raum S2229

## Zusammenfassung

Algorithmen für die Verarbeitung von Texten haben eine lange Geschichte in der Informatik.
Probleme, in denen Texte verarbeitet werden müssen, tauchen nicht nur im Zusammenhang mit Textverarbeitungssystemen (z.B. den bevkannten UNIX-Programmen `grep`, `awk`, `emacs`, `vi`, `compress` und `zip`) auf, sondern kommen in Bereichen wie der Theorie der formalen Sprachen, beim Übersetzerbau, bei der Implementierung von Programmiersprachen, bei Volltextdatenbanken, bei Multimedia Systemen, der Bildverarbeitung und der Bioinformatik vor — insbesondere die Bioinformatik hat in den letzten Jahren zu einem erneuerten Forschungsinteresse an Textalgorithmen geführt.
Das wohl grundlegendste Problem in der Textverarbeitung ist das *Suchproblem (Pattern Matching)*, in dem es darum geht, zu entscheiden ob, wo und wie oft ein Muster in einem Text vorkommt. Im Seminar werden verschiedene Variationen von Algorithmen zur Lösung dieses Problems betrachtet (*exact*, *approximate*; *off-line*, *on-line*). Dazu kommen Algorithmen zur Analyse und zum Vergleich von Texten (Symmetrien in Texten, „Edit Distance“, Unifikation) und Algorithmen zur Textkomprimierung.

Es wird auf eine klare Darstellung der Probleme und Algorithmen Wert gelegt, wobei auch die Analyse (Korrektheit und Laufzeit) berücksichtigt werden soll. Die zwei letzten Seminarthemen befassen sich mit anspruchsvolleren Analysen des Suchalgorithmus von Boyer-Moore und sind deshalb mit einem Stern gekennzeichnet.
Die meisten Themen stammen aus dem Buch „Text Algorithms“ von Maxime Crochemore und Wojciech Rytter, Kapitel 1 und 2 dieses Buches geben eine gute Übersicht und führen in das Thema ein.

Themenliste

## Pattern Matching (off-line)

1. Algorithmus von Knuth-Morris-Pratt

   [1] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 3

2. Algorithmus von Boyer-Moore

   [2] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 4

## Pattern Matching (on-line)

3. McCreight's Algorithmus

   [3] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 5.1–5.3

4. Weiner's Algorithmus

   [4] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 5.1, 5.2, 5.4

5. On-line Konstruktion von Suffix Bäumen: Ukkonen's Algorithmus

   [5] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 5.1, 5.2, 5.5

## Variationen über Pattern Matching

6. Approximate pattern Matching: Berechnung der Edit Distance und „longest common subsequence"

   [6] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 11.1, 11.2

7. Matching mit „don't care" Symbolen

   [7] R. Baeza-Yates, G. H. Gonnet: A new approach to text searching. Communications of the ACM, 35(10):74–82, 1992

8. Randomisierte Algorithmen, Karp-Miller-Rabin

   [8] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 15

[9] R. Motwani, P. Raghavan: Randomized Algorithms, Cambridge University Press, 1995, Kapitel 7.6

9. Lineare Unifikation

[10] M. Paterson, M. Wegman:Linear Unification. Journal of Computer and System Sciences, 16:158–167, 1978

**Weitere Themen**

10. Pattern Matching Automaten: Aho-Corasick

[11] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 7.1

11. Symmetrien in Texten: wiederholte Teilwörter

[12] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 8

12. Text Kompression: Huffman coding und Lempel-Ziv

[13] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 10

13. Ein Approximationsalgorithmus für „shortest common superstrings“

[14] M. Crochemore, W. Rytter: Text Algorithms, Oxford University Press, 1994, Kapitel 15.3

[15] J. Gallant, D. Maier, J. Storer: On finding minimal length superstrings. Journal of Computer and System Science, 20:50–58, 1980

**Analyse des Algorithmus von Boyer-Moore**

14. *Cole's Analyse von Boyer-Moore

[16] R. Cole: Tight Bounds on the Complexity of the Boyer-Moore String Matching Algorithm. SIAM Journal on Computing, 23(5):1075–1091, 1994

15. *„Average case“ Analyse von Boyer-Moore

[17] R. A. Baeza-Yates, M. Regnier: Average running time of the Boyer-Moore-Horspool algorithm. Theoretical Computer Science, 92(1):19–31, 1992