INSTITUT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Vorlesungsinhalt

Semester:	Wintersemester 2002/03
Vorlesung:	Effiziente Algorithmen und Datenstrukturen (4+2) (mit Übungen)
Dozent:	Prof. Dr. Ernst W. Mayr
Übungsleitung:	Klaus Holzapfel
Texte:	Alfred V. Aho and John E. Hopcroft and Jeffrey D. Ullman: "The design and analysis of computer algorithms" Addison-Wesley Publishing Company: Reading (MA), 1974
	Thomas H. Cormen and Charles E. Leiserson and R.L. Rivest: "Introduction to algorithms" McGraw-Hill, 1990
	Donald E. Knuth: "The art of computer programming. V. 3: Sorting and searching" Addison-Wesley Publishing Company: Reading (MA), 1973
	Kurt Mehlhorn: "Data structures and algorithms 1: Sorting and searching" EATCS Monographs on Theoretical Computer Science Springer-Verlag, 1984
	Kurt Mehlhorn: "Data structures and algorithms 2: Graph algorithms and $\mathcal{NP}$-Completeness" EATCS Monographs on Theoretical Computer Science Springer-Verlag, 1984
	T. Ottmann and P. Widmayer: "Algorithmen und Datenstrukturen" B.I. Wissenschaftsverlag, Mannheim, 1990
	Christos H. Papadimitriou and Kenneth Steiglitz: "Combinatorial optimization: Algorithms and complexity" Prentice-Hall, Englewood Cliffs, NJ, 1982

Vorlesungsinhalt:

0. Einleitung
    0. Administrativa
    1. Einleitendes Beispiel
    2. Kombinatorische Probleme
    3. Maschinenmodelle
    4. Komplexitätsmaße
        - 4.1 Berechnungsmodi
        - 4.2 Ressourcen
        - 4.3 Wachstumsverhalten von Funktionen
    5. Rekursionsgleichungen (in Übung)

I. Höhere Datenstrukturen
    1. Grundlegende Operationen
    2. Suchbäume, höhenbalancierte Bäume
        - 2.1 (a, b)-Bäume
        - 2.2 Rot-Schwarz-Bäume
        - 2.3 AVL-Bäume
        - 2.4 Gewichtsbalancierte Binärbäume, BB[α]
        - 2.5 Splay Trees
            1. Amortisierte Kostenanalyse, Potenzialmethode
            2. Splay Trees
        - 2.6 Hashing
            1. Grundlagen
            2. Hashing-Verfahren
            3. Geschlossenes Hashing, Kollisionsauflösung durch Verkettung
            4. Offenes Hashing
    3. Prioritätswarteschlangen
        - 3.0 Unterstützte Operationen
        - 3.1 Binomialbäume
        - 3.2 Binomial Queues/Heaps
        - 3.3 Fibonacci-Heaps
        - 3.4 Buckets
            1. 1-Level-Buckets
            2. 2-Level-Buckets
        - 3.5 Radix-Heaps

3.6 Eine $O(\log\log N)$ Priority Queue

4. Union-Find-Strukturen
   - 4.1 Gewichtete Vereinigung
   - 4.2 Pfadkompression und gewichtete Vereinigung

II. Grundlegende Algorithmen

1. Selektion und Sortieren
   - 1.1 Einleitung
   - 1.2 Der Blum-Floyd-Pratt-Rivest-Tarjan Selektionsalgorithmus
   - 1.3 Randomisierter Medianalgorithmus
   - 1.4 Eine untere Schranke für die Medianbestimmung
   - 1.5 Eine bessere untere Schranke
   - 1.6 Radix und Bucket Sort
2. Minimale Spannbäume
   - 2.1 Grundlagen
   - 2.2 Kruskal's Algorithmus
   - 2.3 Prim's Algorithmus
     1. Einfache Variante
     2. Schnelle Variante
3. Kürzeste Wege/Pfade
   - 3.1 Grundlagen
   - 3.2 Dijkstra's Algorithmus (mit Fibonacci- bzw. Radix-Heaps)
   - 3.3 Floyd's Algorithmus für das all-pairs-shortest-path-problem
   - 3.4 Der Algorithmus von Bellman-Ford
   - 3.5 Digraphen mit negativen Kantengewichten
     1. Grundlagen
     2. Modifikation des Algorithmus von Floyd-Warshall
     3. Modifikation des Algorithmus von Bellman-Ford
     4. Rekalibrierung, der Algorithmus von Johnson
4. Transitive Hülle
   - 4.1 Grundlagen
   - 4.2 Min-Plus-Matrix-Produkt und Min-Plus Transitive Hülle
   - 4.3 Boolesche Matrixmultiplikation und Transitive Hülle
   - 4.4 Der 4-Russen-Algorithmus für Boolesche Matrixmultiplikation
   - 4.5 Transitive Hülle und DFS
   - 4.6 Ein besserer Algorithmus für apsd in ungewichteten Digraphen
   - 4.7 Matrixmultiplikation nach Strassen

5. Matchings in Graphen
   - 5.1 Grundlagen
   - 5.2 Maximum Matchings in bipartiten Graphen
   - 5.3 Maximum Matchings in allgemeinen Graphen
   - 5.4 Matchings in gewichteten Graphen