Institut für Informatik
der Universität München
Prof. M. Hofmann
Dipl. Inf. Hermann Gruber

SS 2007

27. April 2007

Übungen zur Vorlesung
Effiziente Algorithmen

Blatt 2

Aufgabe P-3: Bestimmen Sie die im besten, schlechtesten und durchschnittlichen Fall benötigte Anzahl an Vergleichen für den folgenden Algorithmus, wenn der Durchschnitt über alle Permutationen der Folge $1, 2, \ldots, n$ genommen wird.

Algorithm 2 Teste, ob Liste sortiert ist

```
function isSorted(array a[1,..., n])
for i=1 to n-1 do
  if (a[i] > a[i + 1]) then
    return false;
  end if
end for
return true;
```

Aufgabe P-4: Die harmonischen Zahlen H_n sind definiert durch $H_n = \sum_{k=1}^{n} \frac{1}{k}$.

- Zeigen Sie, dass für $n \in \mathbb{N}$ gilt $1 + \frac{n}{2} < H_{2^n} < 1 + n$.
- Drücken Sie die Summe $\frac{1}{1} + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \ldots + \frac{1}{2n-1}$ durch harmonische Zahlen aus.
- Vereinfachen Sie die Summe $\sum_{k=1}^{n} H_k$.
- Zeigen Sie, dass $H_n = \Theta(\log n)$ und $2^{H_n} = O(n)$.

Hausaufgaben

Aufgabe H-5: Für welche der folgenden Rekursionsgleichungen kann das asymptotische Wachstum der Lösung $T(n)$ mit Hilfe der *Master*-Methode bestimmt werden? Geben Sie in den Fällen, wo dies möglich ist, möglichst scharfe asymptotische Schranken an. (6 Punkte)

- $T(n) = 8\,T(\frac{n}{2}) + 4n^3 + 2n^2$
- $T(n) = 9\,T(\frac{n}{8}) + n \log n$
- $T(n) = 9\,T(\frac{n}{3}) + n^2 \log n$
- $T(n) = 3\,T(\frac{3n}{5}) + n^2$

Aufgabe H-6: Entwerfen Sie einen Algorithmus für das folgende Problem: für eine Menge M von reellen Zahlen mit $|M| = n$ und eine weitere reelle Zahl r ist zu entscheiden, ob sich r als Summe $r = s + t$ zweier Elemente $s, t \in M$ schreiben lässt. Ihr Algorithmus sollte Laufzeit $\Theta(n \log n)$ haben, zeigen Sie dies. (6 Punkte)

Aufgabe H-7: Sei $A = (a_{ik})_{1 \le i,k \le n}$ eine $n \times n$-Matrix mit reellwertigen Einträgen. Der *Entwicklungssatz von Laplace* für die Berechnung von Determinanten gibt uns die rekursive Formel (gültig für alle k in $\{1, 2, \ldots, n\}$)

$$\det(A) = \sum_{i=1}^{n} a_{i,k} \cdot (-1)^{i+k} \cdot \det(U_{i,k}).$$

Dabei bezeichnet $U_{i,k}$ die Untermatrix, die durch Streichen der i-ten Zeile und der k-ten Spalte basiert.

- Geben Sie einen Algorithmus zur Berechnung von $\det(A)$ in Pseudocode an, der auf dieser rekursiven Definition basiert. Benutzen Sie hierzu die gegebenen Funktionen

  $$\begin{array}{rl} \textbf{int}[\,][\,] & \textbf{UnterMatrix}(\textbf{int}[\,][\,]\ \textbf{matrix}, \textbf{int row}, \textbf{int col}) \\ \textbf{int} & \textbf{Dimension}(\textbf{int}[\,][\,]\ \textbf{matrix}) \end{array}$$

  (liefern die Untermatrix $U_{row,col}$ von *matrix*, bzw. für eine $n \times n$-Matrix die Zahl n)

- Sei $T(n)$ die Anzahl der (Gleitkomma-)Additionen und Multiplikationen, die für eine $n \times n$ Matrix ausgeführt werden (Potenzieren von -1 zählt hierbei nicht als Gleitkomma-Operation). Bestimmen Sie $T(k)$ für $k = 1, 2, 3, 4$.

- Zeigen Sie allgemein, dass $T(n)$ in $\Omega(n!)$ ist.

Hinweis zur Analyse: Stellen Sie z.B. eine Rekurrenz auf. Falls die Struktur der Rekurrenz zu kompliziert ist, unterschätzen Sie $T(n)$ geeignet. Nicht zur Bearbeitung: Dieses Aufgabe wirft die Frage nach effizienteren Algorithmen für dieses Problem auf. (2+2+4 = 8 Punkte)

Abgabe bis Dienstag, 8. Mai, 9:15 Uhr in einer der Vorlesungen oder im dafür vorgesehenen Briefkasten in der Theresienstraße. Für Bedingungen zur Bearbeitung siehe die Webseite zur Vorlesung.