Analyse von rekursiven Algorithmen

Aufgabenblatt 4

16. April 2017

Abstract

Die Aufgabe für diese Abgabe war das Schreiben dreierlei Algorithmen für die Berechnung einer Zeile N des Pascalschen Dreiecks. Für diese Algorithmen sollte eine Aufwandsermittlung durchgeführt werden.

**Aufwandsermittlung durch Einbau eines Zählers**

Im ersten Schritt der Aufwandsermittlung sollte in die Programme ein Zähler eingebaut werden, um den Aufwand des jeweiligen Algorithmus zu ermitteln.

Im Folgenden, die dadurch gewonnenen Daten.

N ist die Problemgröße. Sie repräsentiert die Mächtigkeit der zu verarbeitenden Daten.



Tabelle 1

In Tabelle 1 sind die Operationen der drei Implementierungen pro errechneter Reihe des Dreiecks, aufgeführt.

Es wurden Daten bis zur 20. Reihe erfasst, da danach Overflows entstehen.

Durch die Daten lässt sich schon erahnen, welchen Aufwand die Implementierung haben könnten.

**Erläuterung der Algorithmen:**

Im **rekursiven** Programm wird für jede Zahl in der Reihe, welche berechnet werden muss, eine rekursive Methode aufgerufen. Hier der Pseudocode für diese Methode:

*// bbk = berechneBinomialKoeffizient*

int bbk(zeile, spalte){

**if**( spalte=1 || zeile=spalte )

**return** 1

**else**

**return** bbk(zeile-1, spalte) + bbk(zeile-1, spalte-1)

Die Aufwandsfunktion für die Rekursive Implementierung lautet:

Die **Iterative** Implementierung wird wie folgt umgesetzt:

Es gibt eine Schleife und zwei Listen. In einer der Listen sind die Koeffizienten der vorhergehenden Zeile gespeichert, um die Koeffizienten der folgenden Zeile berechnen zu können. Die Schleife wird N-mal durchlaufen.

Die Aufwandsfunktion für diese Implementierung lautet:

Die erhöhte Geschwindigkeit des **Binomial** Programmes kommt durch die Verwendung zweier Tricks: zum einen wird die Funktion der Binomialkoeffizienten verwendet,

um eine Position der Zeile zu berechnen. Zum anderen machen wir uns die Tatsache zu Nutze, dass jede Zeile ab der Mitte gespiegelt ist und die bis dahin berechneten Zahlen an die jeweils gespiegelte Position geschrieben werden können.

Dies wird durch eine nur bis n/2 laufende for-Schleife umgesetzt,

in der jeweils die Fakultäten ermittelt werden und so direkt in die Gleichung eingesetzt werden können.

Die Aufwandsfunktion der Binomial Implementierung lautet:

Abbildung

In Abbildung 1 sind die in Tabelle 1 erfassten Daten, als Grafik dargestellt.

Zunächst, haben alle, bis zur 4. Zeile des Dreieckes, den gleichen Aufwand.

Ab dort ist ein Unterschiedlicher Verlauf zu erkennen.

Die Rekursive Lösung besitzt einen Exponentiellen Anstieg.

Die Iterative Implementierung hat quadratischen Eigenschaften.

Der Algorithmus mit den Binomialkoeffizienten hat einen Anstieg, der als linear zu erkennen ist.