# Analyse von rekursiven Algorithmen

Aufgabenblatt 4

**Abstract**

*Im Folgenden werden drei Verfahren zur Berechnung der Zeile N des Paskalschen Dreiecks analysiert. Ein Iteratives, ein Rekursives und ein „schnelles“ Verfahren, welches sich den Binomialkoeffizienten zur Hilfe nimmt. Das Rekursive Verfahren ist mit einem Aufwand von O(2^n) das langsamste der drei Verfahren. Die Komplexität beiden anderen Verfahren ist O(N).*

## Verfahren 1 – Iterativ

Das Iterative Verfahren muss in jeder Zeile durch jede Spalte. Um eine neue Zeile berechnen zu können, wird das Paskalsche Dreieck in einem zweidimensionalen Array abgespeichert. Die zu berechnende Zeile nimmt sich dann die Ergebnisse der Zeile zuvor um jede Zahl der Spalte aus 2 Spalten der Zeile zuvor zu berechnen. Die erste Zeile des Dreiecks wird vorinitialisiert.

Der Aufwand dieses Algorithmus = O(N2), da zwei ineinander geschachtelte „for-Schleifen“ bis N laufen. Daraus ergeben sich N\*N also N2 Schritte => **O(N2)**

## Verfahren 2 – Rekursiv

Das rekursive Verfahren geht die Länge der gesuchten Zeile: „N“ mit einer „for“-Schleife durch. Dann wird jeder Wert der Zeile durch die Summe von zwei neuen aufrufen derselben Funktion berechnet.

pascal\_recursive(zeile - 1, spalte)+ pascal\_recursive(zeile - 1, spalte - 1);

So wird für jeden Wert die Rekursive Funktion aufgerufen, welche sich wie ein, bis es den gesuchten Wert ermittelt hat, immer weiter aufteilt. Jeder Aufruf hat somit zwei weitere Aufrufe. Die erste Zeile wird mit „1“ initialisiert, was die Abbruchbedingung für die Rekursive Funktion ist.

M = Momentane Spalte

N = Zeile

M\*(T(N-1, M) +(T(N-1,M-1)+1 = O()

## Verfahren 3 – Schnell

Der Algorithmus berechnet die jeweilige N-te Zeile über den Binomialkoeffizient n über k. Wobei n die Zeile und k die jeweilige Spalte in dem Dreieck darstellt. Die Berechnung n über k sieht wie folgt aus: (𝑛𝑘)= 𝑛!𝑘!∗(𝑛−𝑘)!

Um erneutes Berechnen zu sparen, werden bereits berechnete Fakultäten in einer Arraylist gespeichert. So muss nur eine Schleife mit N Durchläufen verwendet werden. Da das Dreieck symmetrisch ist, muss k nur zur Hälfte einer Zeile Laufen (Länge der zweiten Schleife = N/2)

T(N) = N + N/2 => O(N)

## Grafische Auswertung – Aufwandsanalyse

Die grafische Auswertung zeigt den Aufwand(N) über Zeit(T(N)) der drei verschiedenen Verfahren für N von 0 bis 20. D.h. es wurden alle Zeilen des Paskalschen Dreiecks mit den drei vorliegenden Verfahren von Zeile 0 bis Zeile 20 berechnet.

Der grafische Vergleich, der drei verschiedenen Verfahren zeigt den größten Aufwand(exponentiell) bei dem Rekursiven Verfahren. Wie in „Verfahren 2 – Rekursiv“ beschrieben, war dies auch zu erwarten. Da sowohl “Verfahren 1 – Iterativ“ als auch „Verfahren 3 Binomial“ eine Komplexität O(n) aufweisen, war zu erwarten, dass diese auch einen gleichen grafischen Anstieg haben.

## C:\Users\dry\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Matlab pascal.jpg

# N Rekursiv Iterativ Binomial

1 1 0 1

2 2 4 3

3 4 9 5

4 8 15 6

5 16 22 8

6 32 30 9

7 64 39 11

8 128 49 12

9 256 60 14

10 512 72 15

11 1024 85 17

12 2048 99 18

13 4096 114 20

14 8192 130 21

15 16384 147 23

16 32768 165 24

17 65536 184 26

18 1.3107e+05 204 27

19 2.6214e+05 225 29

20 5.2429e+05 247 30