Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Hanjo Täubig

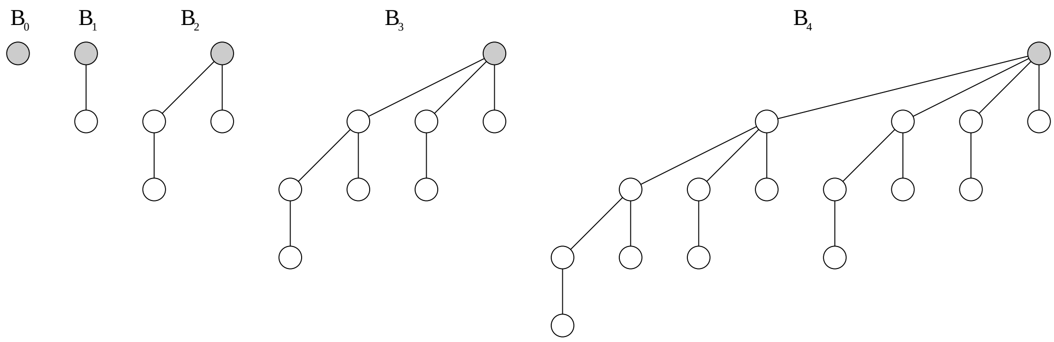
Sommersemester 2006
Übungsblatt 10
7. Juli 2006

Informatik IV

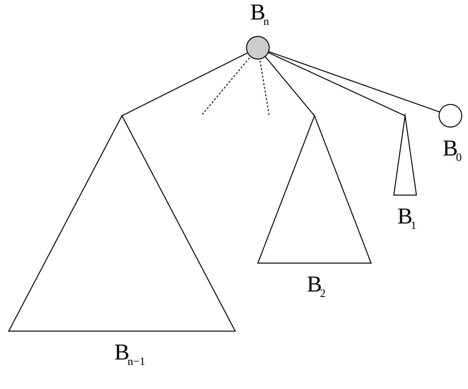
Abgabetermin: 14.07.2006 (vor der Vorlesung)

Aufgabe 1 (10 Punkte)

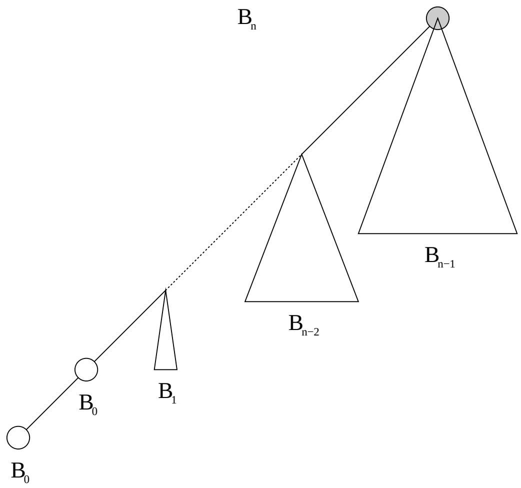
Binomialbäume sind sinnvolle Bestandteile von Datenstrukturen (z.B. für Priority Queues, die in 'Effiziente Algorithmen und Datenstrukturen' behandelt werden). Sie sind rekursiv wie folgt definiert: Der kleinste Binomialbaum B_0 besteht nur aus einem Knoten. Ein Binomialbaum B_k mit $k > 0$ besteht aus zwei B_{k-1}-Bäumen, deren Wurzeln über eine Kante verbunden wurden, so dass die Wurzel des einen B_{k-1}-Baums zum neuen linkesten Kind der Wurzel des anderen B_{k-1}-Baums gemacht wurde.
Zur Erläuterung sind hier die Binomialbäume B_0 bis B_4 dargestellt (die Wurzel ist jeweils schattiert dargestellt):



a) Zeigen Sie, dass die Wurzel des Binomialbaums B_n genau n Kinder hat und dass diese von links nach rechts jeweils die Wurzeln eines B_k für $k = n-1, \ldots, 0$ sind. Das entspricht folgender Zerlegung:



b) Zeigen Sie, dass sich jeder Binomialbaum B_n auch wie folgt entlang eines Rückgrats zerlegen läßt:



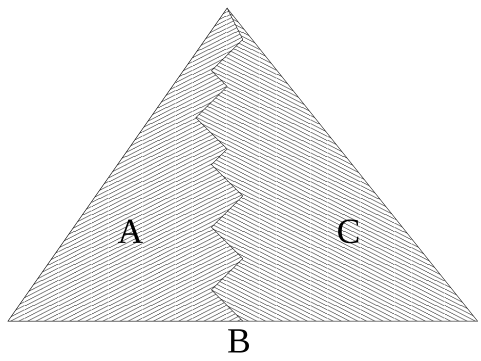
Aufgabe 2 (10 Punkte)

Sei die Pfadlänge P_n eines Binomialbaums B_n definiert als die Summe der Level aller seiner Knoten. Leiten Sie eine geschlossene Formel für P_n in Abhängigkeit von n her.

Aufgabe 3 (10 Punkte)

Gelten folgende Eigenschaften für binäre Suchbäume?

(a) Wir nehmen an, dass ein Pfad von der Wurzel zu einem Blatt gegeben ist. Sei B die Menge aller Knoten, die auf diesem Pfad liegen, sei A die Menge aller Knoten, die links von dem Pfad liegen und sei C die Menge aller Knoten, die rechts von dem Pfad liegen. Gilt für alle $a \in A$, $b \in B$ und $c \in C$, dass $key(a) < key(b) < key(c)$?



(b) Falls wir die Menge B so modifizieren, dass B nur noch aus dem Blatt am Ende des Pfades besteht, gilt die Eigenschaft hier?

Aufgabe 4 (10 Punkte)

Gegeben sei eine Matrix $A \in \mathbb{N}^{m \times m}$ zu der die n-te Potenz A^n berechnet werden soll. Geben Sie einen effizienten Algorithmus dazu an und schätzen Sie die Laufzeit im uniformen und logarithmischen Kostenmaß ab.