Wintersemester 2012/13
Prof. Dr. S. Hougardy
Dr. U. Brenner

Forschungsinstitut für Diskrete Mathematik
Universität Bonn

Einführung in die Diskrete Mathematik
5. Übung

1. Seien (V, F_1) und (V, F_2) zwei Wälder mit $|F_1| < |F_2|$. Man beweise, daß es eine Kante $e \in F_2 \setminus F_1$ gibt, so daß $(V, F_1 \cup \{e\})$ ein Wald ist. (4 Punkte)

2. (a) Zeigen Sie, daß es Folgen von Heap-Operationen gibt, so daß in einem Fibonacci-Heap die maximale Pfadlänge in einer Arboreszenz $\Theta(n)$ ist, wenn n die Zahl der Elemente ist.

   (b) Zeigen Sie, daß zwei Fibonacci-Heaps mit n_1 und n_2 Elementen in $O(\log(n_1 + n_2))$ Zeit verschmolzen werden können. Das Ergebnis soll also ein Fibonacci-Heap sein, der alle $n_1 + n_2$ Elemente enthält. (4 Punkte)

3. Wir wollen eine Menge von Objekten verwalten. Jedem Objekt ist ein Schlüssel zugeordnet, und keine zwei Objekte haben denselben Schlüssel. Außerdem gibt es eine lineare Ordnung auf allen Schlüsseln. Insbesondere wollen wir effizient ein Element mit einem gegebenen Schlüssel suchen. Mit einem sortierten Array geht das durch binäre Suche offenbar in logarithmischer Zeit. Zusätzlich wollen wir aber auch effizient neue Elemente einfügen, was bei einem sortierten Array im allgemeinen lineare Laufzeit erfordern würde. Betrachten Sie statt dessen die folgende Datenstruktur:

   Wir speichern die Objekte in einer Menge von sortierten Arrays, die alle unterschiedliche Länge haben. Die Länge eines jeden Arrays soll eine Zweierpotenz einer ganzen Zahl sein. Jedes Element der Menge soll in genau einem Array vorkommen.

   (a) Zeigen Sie, daß es stets eine solche Zerlegung in Arrays gibt und daß die Längen der Arrays eindeutig sind.

   (b) Beschreiben Sie, wie man ein Element mit gegebenem Schlüssel in Zeit $O(\log^2 n)$ finden kann, wenn n die Zahl der eingefügten Elemente ist.

   (c) Zeigen Sie, daß man, wenn man mit einer leeren Menge beginnt, für eine Folge von n Einfügeoperationen Zeit $O(n \log n)$ benötigt. (4 Punkte)

4. Sei G ein zusammenhängender ungerichteter Graph mit Kantengewichten $c : E(G) \to \mathbb{R}_{\geq 0}$. Dabei seien alle Kantengewichte verschieden, also $c(e) \neq c(e')$ für $e \neq e'$.

   (a) Zeigen Sie, daß es dann genau einen kostenminimalen Spannbaum T in G gibt.

   (b) Ein zweitbester Spannbaum sei ein Spannbaum, der von T verschieden ist und unter allen von T verschiedenen Spannbäumen kleinste Kosten hat. Zeigen Sie, daß es mehrere zweitbeste Spannbäume geben kann.

   (c) Geben Sie einen möglichst effizienten Algorithmus zur Berechnung eines zweitbesten Spannbaums zu einem gegebenen Spannbaum T an. Zeigen Sie die Korrektheit Ihres Verfahrens. (4 Punkte)

Abgabe: Donnerstag, den 15.11.2012, vor der Vorlesung.