Technische Universität München
Institut für Informatik
Lehrstuhl für Effiziente Algorithmen
Dr. Michal Mnuk

SS 1998
Übungsblatt 3
27. Mai 1998

Grundlegende Algorithmen

Abgabe bis 3. Juni 1998 in der Übung

Aufgabe 7

a) In der Vorlesung wurde eine Version von CREATEHEAP vorgestellt.
Dabei stehen zu Anfang n (paarweise verschiedenen) ganze Zahlen in beliebiger Reihenfolge in einem Array, das man als zeilenweise abgespeicherten Binärbaum der Tiefe $h := \lfloor \log_2 n \rfloor + 1$ interpretiert.
Dann wird schichtweise von unten her, beginnend in der vorletzten Schicht $(h-1)$, sukzessive die Heap–Eigenschaft hergestellt, indem jedes Element der aktuellen Schicht solange mit seinem jeweils größten Kind vertauscht wird, bis es größer als seine jeweiligen Kinder ist.

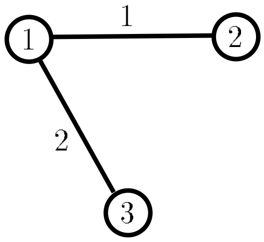
Beweisen Sie formal die Korrektheit des beschriebenen Algorithmus, daß also am Ende jeder Knoten im Baum die Heap–Eigenschaft erfüllt.
(Induktion über ℓ:
i) $\ell = h : \ldots,$
ii) $\ell \to \ell - 1 : \ldots.$)

b) Die Laufzeit des obigen Algorithmus ist, vielleicht erstaunlicherweise, linear. Intuitiv liegt dies wohl daran, daß nur wenige Elemente nahe der Wurzel einen „weiten“ Weg der Länge $\log n$ gehen müssen. Beschreiben Sie in Pseudocode eine naivere und langsamere Version von CREATEHEAP, bei der die Elemente des Eingabearrays einzeln an den Baum angehängt werden und dann, falls nötig, nach oben transportiert werden. Schätzen Sie die asymptotisch (O–Notation) nötige Anzahl von Vergleichen als Funktion von n ab.
Welchen Einfluß hat die Verwendung dieser weniger ausgeklügelten Version auf die asymptotische Laufzeit von HEAPSORT?

Aufgabe 8

Betrachten Sie die in der Vorlesung vorgestellte Version von QUICKSORT, bei der das jeweils erste Element des Arrays als Pivotelement hergenommen wird, und die restlichen Elemente des Arrays in eine linke und eine rechte Hälfte geteilt werden, die jeweils rekursiv sortiert werden.
Sie sollen nun für verschiedenen Eingabefolgen der Länge 5 jeweils einen Graphen mit ebensovielen Knoten zeichnen, der verdeutlicht, welche Elemente in welcher Reihenfolge miteinander verglichen werden. Die Knotennummer i steht für das i–te Element der Eingabefolge. Wenn Element i und j im Ablauf des Algorithmus verglichen werden, soll eine Kante zwischen i und j gezogen werden und fortlaufend numeriert werden.

Beispiel: $(2,3,1) \rightsquigarrow$



Weil zuerst das erste mit dem zweiten, und dann das erste mit dem dritten Folgeglied verglichen wird.

a) $(1,2,3,4,5)$

b) $(4,5,3,2,1)$

c) $(2,5,3,4,1)$

Aufgabe 9

Jemand behauptet, er habe einen vergleichsbasierten RAM–Algorithmus gefunden, der die minimale von n paarweise verschiedene Zahlen mit echt weniger als $(n-1)$ Vergleichen findet.
Beweisen Sie ihm, ohne daß Sie den Code des Programmes anschauen müssen, daß er einen Fehler gemacht hat, weil es nämlich für jedes n mindestens eine Eingabefolge gibt, bei der sein Algorithmus eine falsche Antwort gibt. Geben Sie einen optimalen Algorithmus an.
Tip: Lassen Sie sich von der vorigen Aufgabe inspirieren und verwenden Sie folgendes Ergebnis aus der Graphentheorie:
Ein Graph auf n Knoten und mit echt weniger als $(n-1)$ Kanten kann immer in zwei Teile aufgeteilt werden, die nicht durch Kanten verbunden sind.

Für Interessierte ein Beweis dieses Ergebnisses durch Induktion:
i) $n=2$: Für zwei Knoten ist das trivialerweise richtig.
ii) $n \to n+1$: Angenommen die Aussage stimmt für jeden Graphen G' mit $n-1>2$ Knoten, der höchstens $n-2$ Kanten hat. Sei G ein beliebiger Graph mit $n+1$ Knoten und höchstens $n-1$ Kanten.

- Wenn einen Knoten in G gibt, von dem gar keine Kanten ausgehen ist die Aussage offenbar richtig.
- Anderenfalls gilt: Wenn von jedem Knoten mindestens zwei Kanten ausgingen, so müßte es insgesamt mindestens n Kanten geben, denn jede Kante deckt maximal zwei Knoten ab. Es gibt aber nur höchstens $n-1$ Kanten in G und deshalb gibt es eine Knoten v aus G, von dem genau eine Kante ausgeht. Diese Kante kann aber v nur mit einem der beiden voneinander isolierten Teile von G ohne v verbinden, die es ja nach Induktionsannahme geben muß, denn G ohne v hat n Knoten und höchstens $n-2$ Kanten.