Fakultät für Informations- und Kognitionswissenschaften

Wilhelm-Schickard-Institut für Informatik
Abteilung Rechnerarchitektur

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Scheinklausur

Algorithmen (Prof. Dr. A. Zell), 14.07.2005

Vorname: .... Name: ....

Matrikelnummer: .... Studienrichtung: ....

Semester: ....

Hinweise zur Bearbeitung

- Zählen Sie ihre Aufgabenblätter (7 Seiten inkl. Deckblatt).
- Lassen Sie die Aufgabenblätter zusammengeheftet!
- Schreiben Sie die Lösungen direkt auf die Aufgabenblätter in den dafür vorgesehenen Freiraum.
- Wenn Sie zusätzliche Blätter benötigen, muß jedes davon mit Namen und Matrikelnummer versehen werden. Beginnen Sie jede Aufgabe auf einem neuen Blatt!
- Die alleinige Angabe von Endergebnissen genügt nicht, der Lösungsweg muß erkennbar sein.
- Es sind keine Hilfsmittel, wie z. B. Taschenrechner zugelassen.

Aufgaben-Nummer	maximale Punktzahl	erreichte Punktzahl
1	20	
2	20	
3	20	
Bonuspunkte (Übungen)		
Gesamtpunktzahl		
Note		

Aufgabe 1 Suchen und Sortieren

(a) Geben Sie einen Algorithmus zur Bestimmung des dritt-kleinsten Elements eines unsortierten Feldes $A[1 \ldots n]$ mit Laufzeit $O(n)$ an.

4 Punkte

(b) Geben Sie die genaue mathematische Definition von $g(n) = O(f(n))$ an.

2 Punkte

(c) Geben Sie einen Algorithmus zum Sortieren von Briefen nach 5-stelligen Postleitzahlen an, der in der asymptotischen Zeitkomplexität (O-Notation) schneller als Heapsort ist.

3 Punkte

(d) Konstruieren Sie eine Eingabefolge von $n = 10$ Zahlen $\{1, \ldots, 10\}$, für die Quicksort bei Wahl des jeweils kleinsten Index als Pivotelement-Index maximale Laufzeit hat. Stellen Sie die Rekursionsgleichung für die Zahl der Schlüsselvergleiche auf und lösen Sie diese auf.

3+2 Punkte

(e) Geben Sie eine Schlüsselverteilung 10 ganzzahliger Schlüssel im Bereich $0 \ldots 10^6$ an, bei denen Interpolationssuche deutlich schlechter ist als binäre Suche. Begründen Sie Ihre Konstruktion!

2 Punkte

(f) Heapsort: Geben Sie den Heap an, der nach der Folge von Insert-Operationen der Schlüssel $2, 5, 3 \overset{t_1}{;} 7, 8, 1 \overset{t_2}{;} 6, 4, 9 \;{}^{t_3}$ bei Heapsort entsteht, wenn jedes neue Element zuerst linksbündig eingefügt wird und dann die Heap-Eigenschaft wiederhergestellt wird. Zeichnen Sie die Zustände des Heaps zu den Zeitpunkten t_1, t_2, t_3 in Baumform.

4 Punkte

Aufgabe 2 Bäume

(a) Konstruieren Sie einen möglichst kleinen AVL-Baum, der kein BB[0.3]-Baum ist.

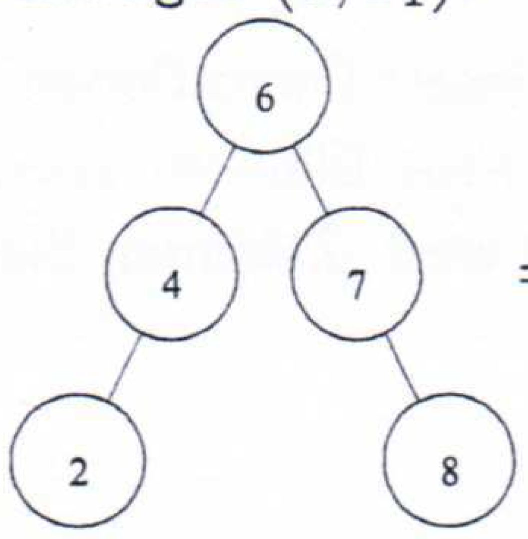
3 Punkte

(b) Konstruieren Sie einen möglichst kleinen BB[0.25]-Baum der kein AVL-Baum ist.

3 Punkte

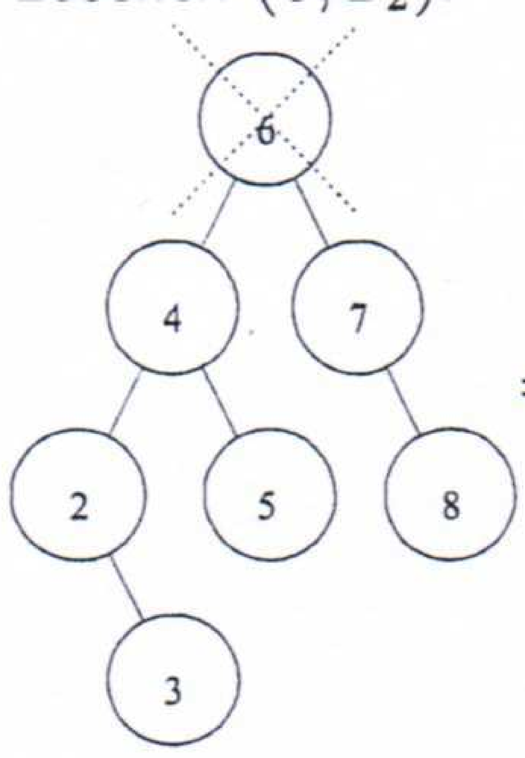
(c) Seien T_i binäre BB[θ.3]-Bäume. Welche Operation ist nach den genannten Einfügen bzw. Löschen zur Rebalancierung nötig und wie sieht der rebalancierte Baum aus?

(i) Einfügen $(3, T_1)$.



$\Longrightarrow$ Operation .... $\Longrightarrow$

(ii) Löschen $(6, T_2)$.



$\Longrightarrow$ Operation .... $\Longrightarrow$

2+3 Punkte

(d) B-Bäume: Bilden Sie einen B-Baum der Ordnung $k = 2$ (nach Def. aus dem Nachtrag) für die Schlüsselmenge 2,4,6,8,10,12,14,16,18,20. (Ein B-Baum hat Ordnung k, wenn jeder innere Knoten außer der Wurzel mindestens $k+1$ Kinder und höchstens $2k+1$ Kinder hat.)

4 Punkte

(e) Bestimmen Sie einen optimalen binären Suchbaum T für die Menge $S = \{1, 2, 3, 4, 5\}$ mit der Zugriffsverteilung $1/15(0, 1, 0, 2, 0, 3, 0, 4, 0, 5, 0)$. Geben Sie seine normierte gewichtete Weglänge P^T an. Nach welcher Algorithmen-Entwurfsstrategie konstruieren Sie Ihre Lösung?

	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$
$k=0$	$r_{11} =$ $p_{11} =$	$r_{22} =$ $p_{22} =$	$r_{33} =$ $p_{33} =$	$r_{44} =$ $p_{44} =$	$r_{55} =$ $p_{55} =$
$k=1$	$r_{12} =$ $p_{12} =$	$r_{23} =$ $p_{23} =$	$r_{34} =$ $p_{34} =$	$r_{45} =$ $p_{45} =$	
$k=2$	$r_{13} =$ $p_{13} =$	$r_{24} =$ $p_{24} =$	$r_{35} =$ $p_{35} =$		
$k=3$	$r_{14} =$ $p_{14} =$	$r_{25} =$ $p_{25} =$			
$k=4$	$r_{15} =$ $p_{15} =$				

4+1 Punkte

Aufgabe 3 Graphen und Zeichenketten

(a) Gegeben sei ein gerichteter Graph $G = (V, E)$ und zwei Mengen von Knoten $A \subseteq V$ und $B \subseteq V$. Geben Sie einen Algorithmus an, der die Länge des kürzesten Weges von einem Knoten $a \in A$ zu einem Knoten $b \in B$ findet. a und b sind nicht vorgegeben, sondern müssen vom Algorithmus bestimmt werden.

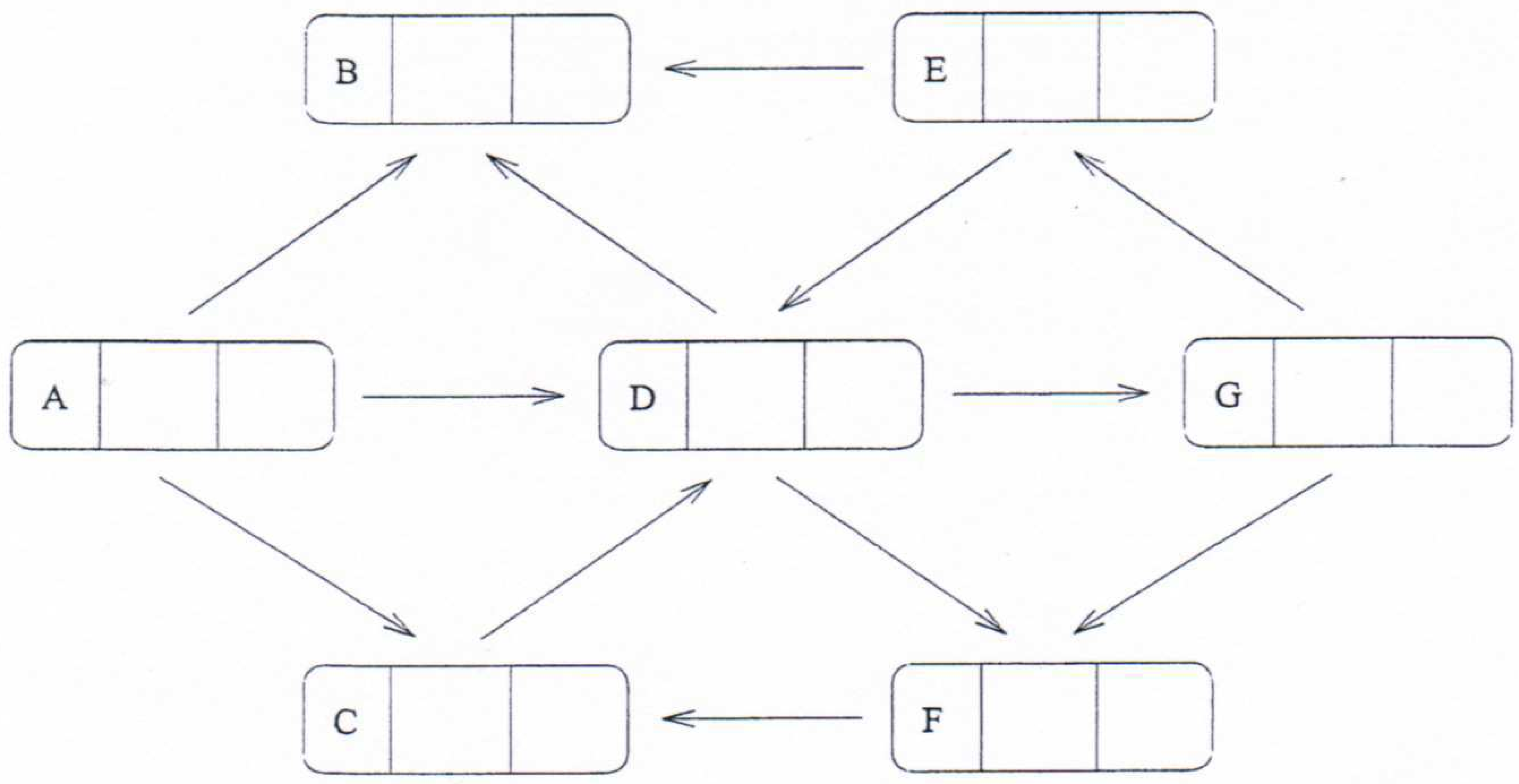
5 Punkte

(b) Geben Sie einen Beispielgraphen mit maximal 5 Knoten an, bei dem der Standard-Dijkstra Algorithmus aus der Vorlesung nicht funktioniert.

3 Punkte

(c) Aufspannender Baum: Bestimmen Sie mit dem modifizierten DFS-Algorithmus der Vorlesung einen aufspannenden Baum für folgenden gerichteten Graphen. Besuchen Sie unter den Nachfolgern zuerst den Knoten mit kleinerer Knotennummer (Buchstabe). Markieren Sie alle Baumkanten als fette Kanten. Schreiben Sie F, B, C an die Forward-, Backward-, Cross-Kanten. Tragen Sie in die Knoten die $dfsnum$ und $compnum$ ein.

5 Punkte



(d) Bestimmen Sie die Präfix Funktion π für "abcababca" des KMP-Algorithmus.

a	b	c	a	b	a	b	c	a

3 Punkte

(e) Zeigen Sie wie der BM-Algorithmus funktioniert, indem Sie den String RING in dem folgenden String suchen.

D	I	E	S		I	S	T		E	I	N		S	T	R	I	N	G
R	I	N	G															

Wie oft wird $T[s+j]$ mit $P[j]$ verglichen?

3+1 Punkte