

Datastructuren (INFOODS)

8 juni 2004

Opgave 1

(1.5 punt)

Gegeven zijn een verzameling S van natuurlijke getallen met de gebruikelijke ordening, en een binaire zoekboom (binary search tree) T op de elementen van S . We hebben een algoritme $GeefPad(T, v)$ dat voor een gegeven knoop v van T de elementwaarden in de knopen op het pad van v naar de wortel van T in een List L zet. Daarbij zal de elementwaarde van de ouder (parent) van v , gevolgd door de elementwaarde van de ouder van de ouder van v , enzovoort. De elementwaarde van de wortel van T zal dus het laatste element van L zijn.

- Kan $L = (42, 42, 42, 42, 42, 43, 17)$ een geldige uitvoer zijn van $GeefPad(T, v)$? Licht je antwoord toe.
- Kan $L = (75, 76, 77, 67, 57, 90, 58)$ een geldige uitvoer zijn van $GeefPad(T, v)$? Licht je antwoord toe.
- Kan $L = (36, 42, 38, 41, 22, 13, 12)$ een geldige uitvoer zijn van $GeefPad(T, v)$? Licht je antwoord toe.

Opgave 2

(2 punten)

Gegeven is een Heap H met n integer keys. Geef een algoritme $KleinGenoeg(H, q)$ dat een List oplevert met alle keys uit H die kleiner zijn dan de gegeven integer waarde q . Zorg ervoor dat de looptijd van je algoritme $O(k)$ is, waarbij k het aantal elementen van de opgeleverde List is. Maak voor toegang tot Heaps en Lists in je algoritme gebruik van de daarvoor beschikbare standaardoperaties (ofwel methods). Toon aan dat je algoritme voldoet aan de gestelde tijdsgrenzen.

Opgave 3

(1.5 punt)

Gegeven is een complete binaire boom T . Bewijs dat voor elke knoop v van T geldt dat de som van de dieptes van alle voorouders (ancestors) van v $O((\log n)^2)$ is.

Opgave 4

(2 punten)

Gegeven is een nette (propere) binaire boom T . We noemen een interne knoop w een *evenwichtsknoop* als het aantal knopen in de deelboom van T die het linkerkind (left child) van w als wortel heeft gelijk is aan het aantal knopen in de deelboom van T die het rechterkind (right child) van w als wortel heeft. We stellen daarnaast dat elke externe knoop een *evenwichtsknoop* is. Geef een zo efficiënt mogelijk algoritme $TelEvenwichtsknopen(T, v)$ dat telt hoeveel evenwichtsknopen er zijn in de deelboom van T die v als wortel heeft. Analyseer de looptijd van je algoritme.

Opgave 5

(1.5 punt)

Gegeven is een hashtable met 15 elementen, en een bijbehorende hashfunctie $h(k) = (k + 4) \bmod 15$. Conflicten (collisions) worden afgehandeld door middel van kwadratische probing.

- a) We hebben achtereenvolgens de waarden 22, 33, 48, 14, 28, 43, 17, 21, 10. Hoe ziet de hashtabel er nu uit?
- b) We zoeken nu of de waarde 13 voorkomt in de hashtabel. Na hoeveel stappen kunnen we concluderen dat 13 niet voorkomt? Licht je antwoord toe.
- c) We verwijderen nu eerst de waarde 14, en voegen daarna achtereenvolgens de waarden 12 en 25 toe. Hoe ziet de hashtabel er nu uit?

Opgave 6

(1.5 punt)

Gegeven is een binaire boom T . We noemen een knoop w een *linkerblad* wanneer w een externe knoop is en bovendien een linkerkind (left child) is van zijn ouder (parent). We noemen een knoop w een *rechterblad* wanneer w een externe knoop is en bovendien een rechterkind (right child) is van zijn ouder (parent). Laat $LB(T)$ het aantal linkerbladeren in T zijn, laat $RB(T)$ het aantal rechterbladeren in T zijn, en laat $I(T)$ het aantal interne knopen in T zijn.

- a) Bewijs dat $RB(T) \leq I(T)$.
- b) Geldt ook dat $LB(T) \leq I(T)$? Licht je antwoord toe.
- c) Geldt ook dat $LB(T) + RB(T) \leq I(T)$? Licht je antwoord toe.