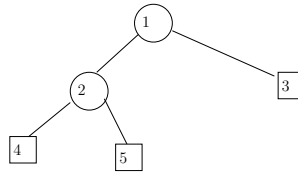


tentamen datastructuren 20 december 2007

Opgave 1.

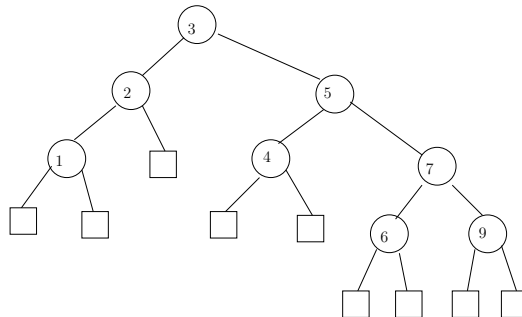
- (a) Beschrijf (hoeft niet in pseudo-code) hoe je de operaties enqueue en dequeue van het ADT voor queues kunt implementeren met twee stacks S_1 en S_2 . (5 punten)
- (b) Beschrijf (hoeft niet in pseudo-code) hoe je een priority queue kunt gebruiken om een stack van grootte N te implementeren. (5 punten)
- (c) Level-order traversal van een (echte) binaire boom bezoekt alle knopen van de boom laag voor laag van boven naar beneden, en per laag van links naar rechts, zoals aangegeven in het plaatje:



Beschrijf (hoeft niet in pseudo-code) een algoritme voor level-order traversal.

(6 punten)

Opgave 2. Gegeven is de volgende AVL-boom:



- (a) Geef stap voor stap aan hoe een knoop met label 8 toegevoegd wordt. (5 punten)
- (b) Geef stap voor stap aan hoe de knoop met label 2 verwijderd wordt. (Werk met de gegeven boom; niet met het resultaat van 2(a).) (5 punten)

Opgave 3.

- (a) Leg uit hoe het kan dat een sorteer-algoritme A in $O(n^2)$ voor sommige inputs sneller is dan een sorteer-algoritme B in $O(n \log n)$.
(5 punten)
- (b) Gegeven is het volgende algoritme:

```
Algorithm Loop(n):  
s ← 0  
for i ← 1 to n do  
  for j ← 1 to 2i do  
    s ← s + i
```

Geef een grote- O karakterisering van de tijdscomplexiteit van Loop.
(5 punten)

Opgave 4.

- (a) Geef de quick-sort boom voor het uitvoeren van quick-sort op de volgende input (de pivot is steeds het laatste element):

1 9 8 5 2 7 4 3 6

- (5 punten)
- (b) Een *inversie* in een rij getallen is een tweetal i en j uit die rij zodat i vóór j komt en $i > j$.
Beschrijf (hoeft niet in pseudo-code) een algoritme dat als input een rij neemt en als output geeft het aantal inversies van die rij.
(6 punten)
- (c) Wat is de tijdscomplexiteit van het algoritme uit 4(b)?
(3 punten)

Opgave 5. Deze opgave gaat over het samenvoegen van lijsten. We hebben een algoritme A dat als input twee lijsten neemt, en als output geeft de concatenatie van die twee lijsten. Het kost $n + m$ stappen om A uit te voeren op twee input-lijsten ter lengte n en m .

- (a) Gegeven zijn vier lijsten, L_1 , L_2 , L_3 , en L_4 met lengte respectievelijk 10, 20, 30, 40.

Wat is de optimale (goedkoopste) manier om gebruik makend van A de vier lijsten samen te voegen tot één lijst en hoeveel kost dat?

(6 punten)

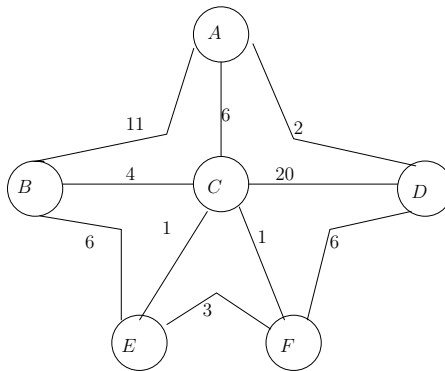
- (b) Beschrijf (hoeft niet in pseudo-code) een greedy algoritme dat gebruik maakt van A voor het algemene geval: de input bestaat uit k lijsten L_1, \dots, L_k en de output is een concatenatie van die lijsten.

Het doel is dat de kosten voor het samenvoegen van die k lijsten zo laag mogelijk zijn.

(6 punten)

Opgave 6.

- (a) Beschrijf stap voor stap het toepassen van Dijkstra's kortste pad algoritme met als input de volgende ongerichte gewogen graaf en startpunt A :



(8 punten)

- (b) Maak van een gerichte graaf met ook negatieve gewichten een gerichte graaf met alleen positieve gewichten door bij elk gewicht n op te tellen als het kleinste gewicht $-n$ is. (Dus als het kleinste gewicht -4 is tel je overal 4 bij op; een gewicht van 3 wordt dan een gewicht van 7.)

Laat zien met een voorbeeld dat Dijkstra's kortste pad algoritme uitgevoerd op de aangepaste graaf (die met de positieve gewichten) in het algemeen *niet* werkt om kortste paden in de oorspronkelijke graaf (die met ook negatieve gewichten) te vinden.

(5 punten)

Opgave 7. Gegeven zijn het patroon P en de tekst T :

$$\begin{aligned} P &= ababc \\ T &= ababddcababc \end{aligned}$$

- (a) Hoeveel stappen voert het brute-force pattern matching algoritme uit?
(5 punten)
- (b) Geef de last functie voor het patroon P .
(5 punten)
- (c) Beschrijf stap voor stap het toepassen van het Boyer-Moore pattern matching algoritme.
(5 punten)

Het cijfer is (het totaal aantal punten plus 10) gedeeld door 10.