

Schriftelijk tentamen Datastructuren

Ma 8 aug 2011 10-13 uur

Met voorbeeld uitwerking

1. Gegevens kunnen expliciet of impliciet voor computerbewerkingen worden opgeslagen; leg uit aan de hand van een vierkant digitaal beeld van 256 grijswaarde pixels die elk eenmaal in random volgorde worden ingevoerd (elke pixel komt binnen met een 2D positie en een helderheid):

a) Hoe bespaard kan worden op de opslag van dit digitale beeld door een datastructuur te kiezen waarbij bepaalde gegevens expliciet en bepaalde gegevens impliciet worden opgeslagen.

b) Als bij de invoer zowel een byte voor de positie horizontaal als die voor vertikaal en een byte per grijswaarde van het pixel gebruikt wordt, wat is dan de besparing procentueel van de gekozen datastructuur onder a) vergeleken met de grootte van de invoer?

(10 punten)

Antwoord op 1a goed voor max 5 punten:

De plaatscoördinaten van elk pixel liggen op een rechthoekig rooster, waardoor deze impliciet kunnen worden opgeslagen als zijnde gerepresenteerd door de rechthoekige array index i, j die samen met een eventueel opgeslagen ware dx, dy pixelgrootte de fysische plaatscoördinaten kan terughalen.

Expliciet kan op elk array adres de grijswaarde worden opgeslagen; dus $G(x, y) \rightarrow G[i, j]$.

Antwoord op 1b goed voor max 5 punten:

Vierkant digitaal grijswaarde beeld van 256 grijswaarde pixels telt 16×16 posities elk aangegeven met 2 bytes en een grijswaarde die ook 1 byte $[0..255]$ inneemt; invoer in totaal 256×3 bytes.

Voor opslag in een 16×16 byte array met de $G[i, j]$ pixel grijswaardes is slechts 256×1 byte nodig, wat $1/3$ van de aangevoerde informatie kost; besparing dus $2/3$ ofwel 67%.

2. Geef een Array Mapping Functie die de matrix $M[i][j]$ voor opslag van 32bit integer waarden met $i \in [5..10]$ en $j \in [4..12]$ afbeeldt op een lineair stel bytes startend op (byte)adres 500.

(8 punten)

Antwoord op 2 voor max 8 punten:

Elk array element $M[i][j]$ kost 32 bit=4 bytes dus per element adres 4 hoger;

$i \in [5..10]$ steeds 6 stuks; $j \in [4..12]$ steeds 9 stuks; geheel $9 \times 6 = 54$ adressen die $54 \times 4 = 216$ bytes innemen.

2D naar 1D adresvertaling bij rij voor rij opslag:

$M[i, j] \rightarrow (i-5) \times 9 \times 4 + (j-4) \times 4 + 500$ hiermee $M[5, 4] \rightarrow 500$ en $M[10][12] \rightarrow 5 \times 9 \times 4 + 8 \times 4 + 500 = 712$ (t/m 715)

2D naar 1D adresvertaling bij kolom voor kolom opslag:

$M[i][j] \rightarrow (j-4) \times 6 \times 4 + (i-5) \times 4 + 500$ hiermee $M[5, 4] \rightarrow 500$ en $M[10][12] \rightarrow 8 \times 6 \times 4 + 5 \times 4 + 500 = 712$ (t/m 715)

Beide lineariseringens plaatsen de matrix binnen adresrange $[500..715]$ met 216 bytes.

3. Gegeven is een dubbelverbonden circulaire lijst met een Head- en een Current pointer:
- Als de next- en previous pointer van het lijst element waarnaar de Head pointer wijst dezelfde waarde hebben als die waarnaar de Current pointer wijst en voor de next- en previous pointer van het lijst element waarnaar de Current pointer wijst geldt dat die dezelfde waarde hebben als die waarnaar de Head pointer wijst, hoeveel elementen staan er dan nog in deze lijst (teken de situatie eerst).
 - Hoeveel elementen staan er in deze lijst als alle 6 voornoemde pointers dezelfde waarde hebben? (10 punten)

Antwoord op 3a goed voor max 6 punten:

Mogelijke situatie als de lijst nog maar 1 of 2 knopen bevat; Bij 1 knoop zie 3B; bij 2 knopen tekening dubbelverbonden rij van 2 knopen met elk een next- en previous pointer; een van de 2 knopen wordt tevens aangewezen door de Head-pointer, de andere door de Current-pointer.

Antwoord op 3b goed voor max 4 punten:

Als de rij nog uit maar 1 knoop bestaat wijzen Head- en Current-pointer naar deze knoop, maar ook de previous- en next-pointer van deze ene knoop wijzen naar zichzelf.

- Geef aan wanneer een Binaire Zoek Boom (BST) met uniek voorkomende integers voor het opzoeken van een bepaalde integer een complexiteit slechter dan $O(\log N)$ heeft.
- De 4 knopen 6, 18, 33 en 15 worden at random toegevoegd : hoeveel volgordes en hoeveel verschillende binaire zoek bomen zijn hierbij mogelijk?
- Bij welke 2 aanvoervolgordes ontstaan de zogenaamde "backbone" structuren? (10 punten)

Antwoord op 4a goed voor max 3 punten:

Zon BST heeft een opzoekcomplexiteit slechter dan $O(\log N)$ als de boom slecht gebalanceerd is; het slechtst denkbare geval treedt op bij de backbone structuur.

Antwoord op 4b goed voor max 4 punten:

4 knopen 6,18,33,15; hiervoor zijn 4! Ofwel 24 Invoegvolgordes waarvan de eerste 6 gegeven zijn als:

6,15,18,33 -> backbone naar rechts structuur 4 niveaus (steeds rechterkind toegevoegd)

6,15,33,18-> 6 in wortel, 2xrechterkind (15,33); 1xlinkerkind(18 bij 33) 3 niveaus

6,18,15,33-> 6 in wortel, 2xrechterkind(18,33); 1 xlinkerkind (15 bij 18) 3 niveaus

6,18,33,15-> 6 in wortel, 2xrechterkind(18,33); 1xlinkerkind(15 bij 18) 3 niveaus

6,33,15,18-> 6 in wortel, 1xrechterkind(33); 1xlinkerkind met rechterkind (15 met 18) 4 niveaus

6,33,18,15-> 6 in wortel, 1xrechterkind(33); 1xlinkerkind met linkerkind(18 met 15) 4 niveaus

.....

18,6,15,33-> 18 in wortel, 1xlinkerkind(6); 1xrechterkind met rechterkind(15 met 33) 3 niveaus

.....

33,18,15,6-> backbone naar links structuur 4 niveaus (steeds linkerkind toegevoegd)

De eerst aangeboden waarde komt in de wortel, vervolgens wordt links/rechts toegevoegd al naar gelang <>knoopwaarde, aangegeven is resultaat met aantal niveaus vermelding.

Antwoord op 4c goed voor max 3 punten:

Bij de geheel gesorteerde aanvoervolgorde (6,15,18,33 en 33,18,15,6) ontstaan de backbone structuren.

5. Hoe gebalanceerd is een Binaire Zoek Boom (BST) als gegeven is dat:

a) Alle bladeren op hetzelfde nivo zitten.

b) Lokaal voor elke knoop de balans 0 is.

c) Geef de AVL voorwaarden voor een (hoogte) gebalanceerde Binaire Zoek Boom (BST).

d) Geef voor een perfect gebalanceerde Binaire Zoek Boom (BST) met 7 elementen een AVL resultaat dat ook voldaan zou hebben aan de balans voorwaarden, terwijl het resultaat verschilt van de perfect gebalanceerde BST.

(9 punten)

Antwoord op 5a goed voor max 2 punten:

Alle bladeren op hetzelfde niveau komt voor bij zowel de perfect gebalanceerde boom als de backbone structuur; is dus geen goede aanwijzing van mate van gebalanceerd zijn.

Antwoord op 5b goed voor max 2 punten:

Als lokaal de hoogtebalans voor elke knoop 0 is hebben we te maken met een perfect gebalanceerde boom die $2^N - 1$ knopen heeft.

Antwoord op 5c goed voor max 2 punten:

AVL stelt de volgende eisen aan een BST: hoogtebalans in elke knoop 0 of 1 (-1,0,1)

Antwoord op 5d goed voor max 3 punten:

Stel knopen A t/m G:

Wortel:

E

Niv 1:

C

G

Niv 2:

B

D

F

Niv 3:

A

Lokaal is de hoogtebalans hier steeds maximaal 1 (AVL voorwaarde)

6. a. Geef aan hoe in een threaded BST de pointers naar de L en R kind knopen gebruikt worden als er geen kind aanwezig is.

b. Hoe onderscheid je pointers naar kinderen van pointers naar opvolgers?

(7 punten)

Antwoord op 6a goed voor maximaal 4 punten:

Bij een threaded BST wordt in principe het ontbrekende kind link verlegd naar die knoop die in de InOrder doorloop (LOR volgorde) de volgende te bezoeken knoop zou zijn; voor het ontbrekende

linkerkind is dit een link naar de Ouderknoop, voor het ontbrekende rechterkind de volgende knoop die bezocht zou worden bij de InOrder doorloop.

Antwoord op 6b goed voor max 3 punten:

Om pointers naar een kinderknoop te onderscheiden van pointers naar een opvolgerknoop (bij een lege kindknoop) is minimaal 1 bit nodig; soms wordt het tekenbit hiervoor gebruikt (de adresseringsrange wordt daarmee wel gehalveerd).

7. a. Geef de definitie van een minimale perfecte hash functie.

Gegeven de volgende reeks 30 studentID's:

1034367

1045091

1017713

1080644

1052616

1047493

1075152

1023144

1045105

0934615

1021869

1015265

1045113

1047515

1085298

1045121

1075160

1045148

0704598

1068423

1045156

1075659

0748439

1023160

1045164

0719374

1080652

1036718

1023187

1019872

b. Maak de statistiek op van het voorkomen van elk digit (digit is $\in [0..9]$) op de verschillende posities binnen een studenten ID.

- c. Welke digit posities komen het meest in aanmerking voor opname in een hash key gebaseerd op een 2 digit selectie?
- d. Hoeveel botsingen komen er nog voor als je de twee beste digits gebruikt om een hash key $\in [00..99]$ te construeren?
- e. Zijn eventuele botsingen met een simpele lineaire rehash op te lossen?
(12 punten)

Antwoord op 7a goed voor max 3 punten:

Een minimale, perfecte hash functie berekent een unieke index in een lijst die precies even groot is als het aantal onder te brengen datawaarden.

Antwoord op 7b goed voor max 3 punten:

	0	1	2	3	4	5	6	7	8	9	Tot
D1:	4	26	0	0	0	0	0	0	0	0	30
D2:	26	0	0	0	0	0	0	3	0	1	30
D3:	1	4	4	3	10	1	1	3	3	0	30
D4:	2	1	1	3	3	12	1	3	2	2	30
D5:	1	11	2	2	3	2	5	2	2	0	30
D6:	1	6	2	1	3	4	6	2	1	4	30
D7:	2	2	3	4	4	4	2	2	4	3	30

Antwoord op 7c goed voor max 2 punten:

De beste 2 digit posities waarbij de verdeling van $[0..9]$ het meest homogeen is zijn d6 en d7.

Antwoord op 7d goed voor max 2 punten:

Er treden 6 botsingen op (44, 13,15,98,60,52)

Antwoord op 7^e goed voor max 2 punten:

Alle botsingen zijn hier met een lineaire rehash op te lossen zolang er minstens evenveel hash tabel plaatsen zijn als studentID nummers;

b.v. hier bij 30 ID's en 100 plaatsen en verwijzing naar eerste vrije entry erboven lossen we de 6 botsingen op met 44->45; 13->14;15->17 (16 ook bezet); 98->99;60->61; 52->53.

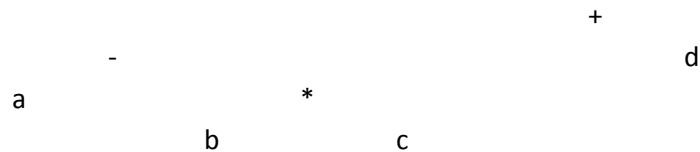
8. Gegeven is de volgende expressie $((a-(b*c))+d)$ die in een bijbehorende expressieboom kan worden ondergebracht:

- Teken de bijbehorende expressieboom
- Geef aan in welke volgorde de expressie elementen opgeleverd worden bij PostOrder doorloop.
- Geef aan in welke volgorde de expressie elementen opgeleverd worden bij InOrder doorloop.
- Wat is de Prefix notatie van deze expressie?

(10 punten)

Antwoord op 8a goed voor max 3 punten:

Tekening expressieboom



Antwoord op 8b goed voor max 3 punten:

PostOrder: abc*-d+

Antwoord op 8c goed voor max 2 punten:

InOrder: a-b*c+d

Antwoord op 8d goed voor max 2 punten:

Prefix: +-a*bcd

9. Bespreek waarom er voor data opslag op harde schijven B-bomen gebruikt worden i.p.v. Binaire Zoek Bomen (BST's) en wat bepaalt hoeveel data er in 1 knoop of blad terechtkomen.

(10 punten)

Antwoord op 9 goed voor max 10 punten:

B-Bomen zijn ontwikkeld voor opslag in files op hard disks; op harde schijven is de laagste communicatie(overdracht) hoeveelheid het aantal bytes dat in 1 blok past (bij formatteren bepaald op 512bytes of veelvoud daarvan zoals 4KB); de B-bomen proberen zodoende voldoende (minstens halfvol blok) waarden in 1 blad(blok) te plaatsen en alle data op blad niveau te houden, zodat op hogere niveaus alleen range indexen staan en er eventueel extra pointers kunnen worden toegevoegd om van blad naar volgend blad (blok na blok) in de sortering te springenvoor een lineaire doorloop.

10. Geef voor de volgende spellen gebaseerd op gelijkzijdige 3-, 4- of 6-kanten (Triominos, Scrabble, Catan b.v.) elk met zijdes van 1 cm van hun 2D grid structuur:

a) Van elk de vorm en grootte in cm^2 van de bijbehorende "space filling" gridcel.

b) Van elk het aantal (connectiviteit) en de afstand van hun dichtstbijzijnde bureu.

c) Welk van deze 3 gridcellen is het best geschikt om als recursief opdeelbaar beschrijvings-element van het aardoppervlak te dienen (geef reden, initiële opdeling en teken de recursieve opdeling van 1 zo'n element).

(12 punten)

Antwoord op 10a goed voor max 4 punten:

Voor de roostercellen zelf geldt de oppervlakte horend bij de gelijkzijdige veelhoek met zijde 1;

Voor de driehoek is dat $\frac{1}{2} * \sqrt{3}/2 = \sqrt{3}/4$

Voor het vierkant $1 \times 1 = 1$

Voor de zeshoek is dat 6x oppervlak van een gelijkzijdige driehoek is $6 \cdot \frac{\sqrt{3}}{4} = 3\frac{\sqrt{3}}{2}$

Gridcel is de vorm van de ruimte tussen zwaartepunten van de regelmatige veelhoeken in:

Bij gelijkzijdige driehoeken is dat een regelmatige zeshoek met zijde $\frac{\sqrt{3}}{3}$ en oppervlak $\frac{\sqrt{3}}{4}$

Bij vierkanten is dat eenzelfde vierkant met zijde 1 en oppervlak 1

Bij zeshoeken is dat een gelijkzijdige driehoek met zijde $\frac{\sqrt{3}}{2}$ en oppervlak $3\frac{\sqrt{3}}{4}$

Antwoord op 10b goed voor max 4 punten:

Driehoeksrooster: 3 naaste burens op $2\frac{\sqrt{3}}{6} = \frac{\sqrt{3}}{3}$

Vierkantsrooster: 4 naaste burens op afstand 1

Zeshoeksrooster (hexagonaal): 6 naaste burens op afstand $2 \cdot \frac{\sqrt{3}}{2} = \sqrt{3}$

Antwoord op 10c goed voor max 4 punten:

Voor een beschrijving van het aardoppervlak met gekromde roosterzellen is de gelijkzijdige driehoek het eerlijkst, uitgaande van een initiële opdeling passend bij een octaëder wordt een eerste belegging gemaakt in 8 vrijwel gelijke delen door het aardoppervlak bij de evenaar en 2 maal loodrecht daarop bij b.v. 0 en 90 graden lengte op te delen middels 3 onderling loodrechte vlakken -> 8 gekromde driehoeken. Deze driehoeken kunnen verfijnd worden door de middens van zijden te verbinden en zo elke driehoek in 4 kleinere, even grote driehoeken op te delen; dit kan herhaald worden tot de gewenste diepte (eventueel hiërarchisch).