

Examen Algoritmen en Datastructuren III

Naam :

Stellingen uit de les mogen natuurlijk altijd gebruikt worden zonder dat het bewijs opnieuw gegeven moet worden (behalve in gevallen waar het expliciet anders staat)!

1. B+-trees en B-trees (3 pt)

Natuurlijk is een B+-boom met grootte 2 heel onrealistisch, maar omdat voorbeelden voor grotere n heel veel tijd vragen om uitgewerkt te worden, werk nu met grootte 2:

Voeg records met identificatiesleutels 10, 17, 20, 12, 13, 14, 15, 16 in deze volgorde toe aan een B+-boom met grootte 2. Toon voldoende details en tussenstappen om te kunnen zien wat er gebeurt.

- Bewijs dat in een B+ tree elke sleutel ten hoogste twee keer voorkomt – één keer in de bladeren en één keer in een interne top.

- Stel dat in een B-tree met grootte 30 en met n sleutels (waarbij n voldoende groot is) 90 percent van de sleutels een grafsteen hebben – dus passief zijn. Hoe sterk verschilt de bovengrens voor de maximale diepte ten hoogste in vergelijking met de bovengrens voor een B-tree met grootte 30 die alleen maar de actieve toppen bevat? Geef uitleg.

2. Metaheuristieken (2 pt)

- Voor welk soort problemen worden metaheuristieken toegepast? Geef geen voorbeelden maar typische eigenschappen van deze problemen.

- Beschrijf het algoritme van *variable neighbourhood search*.

4. Compressie (2 pt)

Toon voor elke van de volgende oefeningen voldoende tussenstappen en details om te kunnen zien wat er gebeurt.

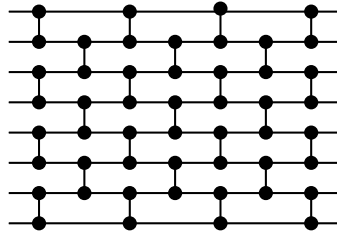
- Decodeer de code 0,9,3,6,5,7,9,4,3,5,8,6,0,9,9. Deze code is het resultaat van een Burrows-Wheeler transformatie en achteraf een move-to-front codering met de beginlijst $\{6, d, e, g, l, o, p, s, t, -\}$. Om de positie van het begin van de tekst aan te geven werd bij de berekening van de Burrows-Wheeler getransformeerde met 0 begonnen te tellen.

- Decodeer de met LZW gecodeerde tekst. De code is
 $g, e, l, i, 256, -, z, e, 263, 256, l, 262, e$
waarbij voor lettertekens die al in het begin in het woordenboek staan (dus met code $0, \dots, 255$) het letterteken in plaats van de code geschreven is.

- Leg precies uit wat het effect van de grootte van het sliding window is als een tekst met LZ77 wordt gecodeerd. Wat is het effect tijdens het decoderen? Moeten voor het coderen en het decoderen sliding windows van dezelfde grootte gebruikt worden? Geef uitleg.

5. Sorteernetwerken/Shearsort (3 pt.)

- (a) Sorteert het onderstaande netwerk correct? Zo nee, geef een tegenvoorbeeld. Zo ja, bewijs.



- (b) Bespreek de werking van het shearshort-algoritme op een rooster van processoren. Bewijs correctheid en complexiteit van het algoritme (gebruik daarbij het nul-een-principe voor sorteernetwerken).

6. Parallele prefixbewerkingen (2 pt.)

Gegeven een rij a van n gehele getallen, die negatief kunnen zijn. Gevraagd is twee indices i en j te bepalen zodanig dat $\sum_{k=i}^j a_k$ zo groot mogelijk is.

Geef een PRAM-algoritme, dat gebruik maakt van parallele prefixberekeningen, om dit probleem in $\Theta(\log n)$ parallele tijd op te lossen.

7. Exact string-matching (2 pt.)

- (a) Geef een voorbeeld van tekst (van lengte n) en patroon (van lengte m) waarvoor het algoritme van Boyer-Moore-Horspool hoogstens $O(n/m)$ vergelijkingen doet.
- (b) Hoeveel vergelijkingen doet het algoritme van Knuth-Morris-Pratt in dit geval?

NOG NIET OMDRAAIEN !