

Typische Examenvragen

Reinoud Berkein

May 26, 2018

- Inleiding
 - Gegeven één of andere tabel met probleemgroottes N en uitvoeringstijden (zie bvb. doubling experiment). Wat kan je besluiten ivm het gedrag van dit specifieke algoritme?
 - Leg het verschil uit tussen grote- O en tilde-notatie om het stijgingsgedrag van functies te karakteriseren.
- Selection Sort, Insertion Sort, Merge Sort
 - Stel dat ik voor een groot aantal verschillende arrays van verschillende lengte N zowel SelectionSort, InsertSort, als MergeSort laat lopen om de arrays te sorteren. Geef een grafiek van de uitvoeringstijden i.f.v. N (schets de grafiek, verklaar de grafiek, verklaar het verloop van de verschillende functies, ...) (Belangrijk: grafieken hebben geen zin als je niet minstens de assen benoemt, en effectieve getallen op de assen plot. Zeker voor vergelijkingen tussen functies moeten onderlinge verhoudingen juist zijn.)
- Quicksort
 - Bewijs dat de tijdscomplexiteit van Quicksort $\sim 1.39n\log(n)$.
 - Indien we Quicksort zouden implementeren met 2 pivots (en 3 partities), hoe zou de tijdscomplexiteit zich gedragen? Let op: dit is niet hetzelfde als een 3-way Quicksort waarbij 1 van de partities enkel elementen bevat die gelijk zijn aan de pivot.
 - De worst-case tijdscomplexiteit van QuickSort is $\sim n^2$, indien men telkens voor elke partitionering, het kleinste dan wel het grootste element als pivot kiest. Men kan echter stellen dat QuickSort vrij vergevingsgezind is, en dat, zelfs al we toevallig enkele malen een slechte pivot kiezen gedurende de hele sortering, we toch

een lineair verloop van de rekestijd kunnen verwachten.
Verklaar! (Examenvraag september 2016)

- Sorteren in lineaire tijd
 - Een aantal sorteeralgoritmes hebben de eigenschap "stabiel" te zijn.
 - * Wat wordt hiermee bedoeld?
 - * Welke sorteeralgoritmes die we behandeld hebben in de lessen zijn stabiel?
 - * Waarom is stabiliteit een belangrijke eigenschap van een sorteeralgoritme?
 - * Kan een niet-stabiel sorteeralgoritme steeds stabiel gemaakt worden?
 - Hoe zou je counting sort kunnen wijigen zodat het ook zou werken met data die geen positieve integers zijn? Bvb ook negatieve integers behandelen? Of ook getallen met decimale fracties (1.5, 2.5, 3.5, ... ?). Welke voorwaarden moeten er gelden voor de te sorteren data?
- Stacks, Queues & Hash Tabellen
 - Zie vraag 8 van examen September 2015 (te vinden in folder van examens vorige academiejaren).
- Priority Queues & Balanced Trees
 - Geef een bespreking van de insert- operaties in een 2-3 boom.
 - Stel dat we ternaire heaps (t.t.z. een heap met 3 kinderen per knoop) zouden gebruiken voor heapsort. Op welke manier zou dit de tijdscomplexiteit beïnvloeden?
 - Vraag 7 examen september 2016 (Young-tabel). Dit is een mooie illustratie hoe heaps niet noodzakelijk in boomstructuren moeten voorgesteld worden, maar bvb. ook een matrix-vorm mogelijk is.
 - Vraag 8, examen juni 2015.
- Greedy Algoritmen
 - Gegeven één of andere string, stel de Huffman coderingsboom op voor deze string.

- Hoe ziet de Huffman codering er uit indien de frequentie van characters zich verhoudt tot de Fibonacci getallen?
- Bewijs dat Huffman-codering optimaal is, t.t.z. de kortst mogelijke gecodeerde string oplevert voor een gegeven te coderen string.
- Minimaal Opspannende Bomen en Kortste Pad
 - Gegeven één of andere grafe, geef de volgorde waarin edges/vertices worden toegevoegd in Kruskal/Prim/Dijkstra/..., of geef de toestand van de priority queue bij Kruskal/Prim/Dijkstra nadat een specifieke vertex toegevoegd is aan de MST of SPT.
 - Leg uit: algoritme van Prim/Kruskal/Dijkstra. Zou je dit algoritme beschouwen als een greedy algoritme (argumenteer), dan wel als een dynamisch (dynamische algoritmen komen aan bod in les 10) algoritme (argumenteer), beide (argumenteer), of geen van beide (argumenteer).
- String Matching
 - Gegeven een string, stel de bijhorende Knuth-Morris-Pratt toestandsmachine op. Zie ook enkele voorbeelden in examenopgaven van vorige jaren.
- Dynamic Programming
 - Zou je het algoritme van Dijkstra beschouwen als een dynamisch algoritme, dan wel een greedy algoritme, of beide? Verklaar?
 - Een variant op het LCS probleem is de langste subsequentie zoeken in een string die tevens een palindroom is. Zie Vraag 7 in examen van juni 2015.