

## Examen Gegevensstructuren & Algoritmen

31 januari 2025

ViS

### Theorie (10 punten, herleid naar 5)

Mondeling (25 min) met schriftelijke voorbereiding (35 min).

#### Vraag 1

- Leg gedetailleerd het QUICKSORT algoritme uit.
- Hoe ziet de gerandomiseerde versie van dit algoritme eruit?
- Wat is de complexiteit in het slechtste, het beste en het gemiddelde geval? Voor het gemiddelde geval volstaat een intuïtieve redenering; een formeel bewijs hoeft niet.
- Wat is de complexiteit wanneer alle elementen van de te sorteren lijst identiek zijn?
- Leg uit hoe je het  $i$ -de kleinste element kan selecteren een (ongesorteerde) lijst van onderling verschillende getallen.

#### Mondelinge bijvragen

- Wat doet PARTITION precies? Leg gedetailleerd uit.
- Hoe kan je de performantie verbeteren ingeval alle elementen van de lijst identiek zijn?  
*Antwoord: splits  $A$  in 3 delen, namelijk een deel waarbij alle elementen  $< A[r]$  zijn, een deel waarbij alle elementen gelijk aan de pivot zijn en een deel waarbij alle elementen  $> A[r]$  zijn. Je hebt naast de variabele  $i$  dus nog een tweede counter  $k$  nodig.*
- Wat is het verschil tussen RANDOMISED-SELECT en QUICKSORT?  
*Antwoord: RANDOMISED-SELECT zal zichzelf enkel recursief aanroepen voor een van de twee helften  $A[p..q-1]$  of  $A[q+1..r]$ ; het QUICKSORT-algoritme doet dat voor beide helften.*

#### Vraag 2

- Geef de definitie van een (binary) heap.
- Beschrijf het algoritme om de structuur van zo'n heap te onderhouden/herstellen.
- Bewijs m.b.v. de mastermethode wat de complexiteit van dit algoritme is.

#### Mondelinge bijvragen

- Waar bevindt het kleinste element zich in een max-heap?  
*Antwoord: in een van de bladeren.*

#### Vraag 3

- Geef 3 technieken die kunnen worden aangewend om te *proben* in een open-address hashtable.
- Bespreek voor elk van deze technieken de voor- en nadelen.

## 1 Oefeningen (10 punten, herleid naar 5)

Hier kreeg je 3 uur de tijd voor.

### Vraag 1 (1 punt)

Los onderstaande recurrentievergelijken op aan de hand van de mastermethode, of leg uit waarom de mastermethode eventueel niet toepasbaar is.

$$T(n) = 16T\left(\frac{n}{4}\right) + n^2\sqrt{n}$$

$$T(n) = 9T\left(\frac{n}{3}\right) + \lg n$$

### Vraag 2 (3 punten)

Gegeven een rij van verschillende positieve gehele getallen  $a_1, a_2, \dots, a_n$ . Er treedt een *inversie* op wanneer voor twee getallen geldt dat  $i < j$  en  $a_i > a_j$ . De rij  $[30, 10, 40, 20]$ , bijvoorbeeld, heeft 3 inversies, die we aanduiden met de posities van hun voorkomen in de rij:  $(1, 2)$ ,  $(1, 4)$  en  $(3, 4)$ .

- Schrijf een  $\Theta(n \lg n)$  algoritme dan het aantal inversies van een gegeven rij berekent.
- Toon de complexiteit van je algoritme aan.

### Vraag 3 (1,5 punten)

Sorteer volgende rij met het counting sort algoritme.

$$A = [5_1, 0_2, 1_3, 0_4, 2_5, 3_6, 5_7, 0_8, 3_9, 2_{10}]$$

Toon voldoende tussenstappen en onderscheid entries met dezelfde sleutel. Het subscript van elk element geeft diens index in de originele rij aan.

### Vraag 4 (2,5 punten)

Normaal gezien worden uitdrukkingen van wiskundige bewerkingen in infix-vorm geschreven, zoals bijvoorbeeld  $9 - 2 * 3 + (4 - 5)$ . Een andere manier om de volgorde van de bewerkingen aan te geven is de *fully parenthesised* schrijfwijze. Het eerder aangehaalde voorbeeld schrijven we dan als  $((9 - (2 * 3)) + (4 - 5))$ .

- Schrijf een zo efficiënt mogelijk algoritme dat een *fully parenthesised* uitdrukking evalueert.
- Bepaal de complexiteit van je algoritme.

Je mag ervan uit gaan dat er geen delingen gebeuren. Denk goed na over de gepaste datastructu(u)r(en) voor je algoritme.

### Vraag 5 (2 punten)

Construeer een rood-zwart boom met sleutels

$$10, 20, 30, 7, 17, 3, 11, 12, 18$$

door deze sleutels in volgorde toe te voegen aan een lege boom.