

Aanwijzingen:

U dient zich te houden aan de stijl van het tekst- en werkboek.

Indien een toelichting wordt gevraagd, wordt deze veelal hoger gewaardeerd dan het overige deel van het antwoord.

Indien de complexiteit van een algoritme wordt gevraagd, moet het antwoord zo precies mogelijk gegeven worden. Als de complexiteit bijvoorbeeld $\Theta(n)$ is, wordt $O(n)$ als antwoord niet geheel goed gerekend. Licht de complexiteit ook altijd toe.

Er zijn in totaal 100 punten te behalen. Het cijfer voor deze serie wordt bepaald in één decimaal.

OPGAVE 1

Gebaseerd op opgave C-4.11 uit het tekstboek (blz. 185).

10 punten

- a Ga er van uit dat polynomen worden opgeslagen in arrays, dat wil zeggen dat coëfficiënt a_i wordt opgeslagen als array-element met index i . Maak nu onderdeel a van C-4.11 en beschrijf de bedoelde methode in pseudocode en licht toe waarom deze $O(n^2)$ is.

5 punten

- b Maak onderdeel b van C-4.11. U hoeft hierbij geen pseudocode te geven maar u moet uw antwoord wel toelichten.

OPGAVE 2

10 punten

- a Maak opgave C-5.4 uit het tekstboek (blz. 218) met de volgende wijziging: in plaats van twee wachtrijen wordt één wachtrij gebruikt. Ga uit van een array-implementatie en beantwoord de opgave zoals opgave 5.2 uit het werkboek.

5 punten

- b Is er bij onderdeel a sprake van een toepassing van het adapter-patroon? Licht toe waarom (niet)?

OPGAVE 3

10 punten

Een array list zullen we meestal implementeren met een array. In bepaalde gevallen is een andere implementatie handiger. Een array list V heet ijl (Eng: sparse) als de meeste elementen van de array list leeg zijn, dwz nul of null, afhankelijk van het soort elementen dat de array list bevat. Als van de n elementen uit de array list er m niet leeg zijn, kunnen we de array list efficiënt opslaan door alleen de m niet-lege elementen op te slaan, samen met het rangnummer van het element in de array list. We gebruiken hiervoor een dubbelgeschakelde lijst. Als de array list het element e met index r bevat, dan slaan we in de lijst het paar (r, e) op. Beschrijf hoe bij deze implementatie de methoden van de Array list ADT uitgevoerd kunnen worden, en bepaal de complexiteit van de methodes bij deze implementatie.

OPGAVE 4

Een min-max heap is een datastructuur waarin het mogelijk is om zowel de operatie deleteMin als de operatie deleteMax in $O(\log n)$ uit te voeren. De structuur maakt net zo als een gewone heap gebruik van een volledige binaire boom. De knooppunten van de boom voldoen aan de volgende twee min-max heap eigenschappen:

- als de diepte van het knooppunt k even is, dan zijn alle sleutels van knooppunten uit de subboom met wortel k groter dan of gelijk aan de

sleutel van dit knooppunt; de sleutel van k is dus minimaal in de bijbehorende subboom.

- als de diepte van het knooppunt k oneven is, dan zijn alle sleutels van knooppunten uit de subboom met wortel k kleiner dan of gelijk aan de sleutel van dit knooppunt; de sleutel van k is dus maximaal in de bijbehorende subboom.

(Voor de volledigheid vermelden we dat de wortel van de heap een even diepte heeft, namelijk 0.)

5 punten

a Teken een min-max heap voor de sleutels 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

7 punten

b Geef een algoritme om een nieuwe knoop aan een min-max heap toe te voegen.

8 punten

c Geef algoritmes voor deleteMin en deleteMax

OPGAVE 5

10 punten

Beschouw 2 verzamelingen met gehele getallen, $S = \{s_1, s_2, \dots, s_n\}$ en $T = \{t_1, t_2, \dots, t_m\}$.

Beschrijf twee verschillende manieren om te testen of S een deelverzameling is van T . U mag alle datastructuren uit het boek hierbij gebruiken. Het is niet de bedoeling dat u hier algoritmes in pseudocode of java code geeft, maar een beschrijving van de algoritmen in woorden. Wat is de complexiteit van elk algoritme?

OPGAVE 6

5 punten

a Lees opgave C-13.6 uit het tekstboek.
Teken nu een boom met 8 knooppunten waarbij het centrum excentriciteit 3 heeft.

10 punten

b Maak opgave C-13.6 uit het tekstboek

5 punten

c Wat is de complexiteit van uw algoritme?

10 punten

OPGAVE 7

Maak opgave C12-12 van het tekstboek.