

Algoritmen, Datastructuren en Complexiteit (214020 en 214025)

Bij dit tentamen mag het boek van Baase en Van Gelder worden gebruikt, evenals een uitdraai van de hoorcollegesheets (dit alles zonder eigen aantekeningen).

Bij de opgaven waar om een algoritme wordt gevraagd, geeft u de pseudocode van uw oplossing en een beknopte maar duidelijke uitleg van de werking. Algoritmes zonder duidelijke uitleg worden niet in beschouwing genomen.

Uitspraken die u doet in antwoord op gestelde vragen moeten nauwkeurig worden beargumenteerd.

Er zijn 5 opgaven, waarmee 90 punten behaald kunnen worden. Het tentamenresultaat is (het aantal behaalde punten gedeeld door 10) plus 1.

Vermeld uw naam en de afkorting ADC op ieder los blad. Vermeld ook de werkcollegeleider waar u dit jaar bij was ingedeeld en geef expliciet aan of u beide huiswerkopgaven gemaakt hebt.

Veel succes!

Opgave 1

15 pt

Beschouw het volgende algoritme (met * vermenigvuldigen, div integer division (bv. $7 \text{ div } 2 = 3$), en 2 kwadraat):

```
int func(int n)
{ if n == 0 return 1
  else if n < 8 return n
    else return 3*func(n div 8) + 8 + func(n div 8)^2
}
```

1. Geef een recursieve uitdrukking van de tijdscomplexiteit van dit algoritme, uitgedrukt in het aantal rekenkundige operaties.
2. Wat is de complexiteitsklasse van dit algoritme?

Opgave 2

20 pt

Stel we hebben een maxheap met n verschillende elementen (gegeven als een array) en $n > 2$. We weten toevallig dat het kleinste element van deze heap index k heeft.

1. Geef een zo efficiënt mogelijk algoritme (qua aantal vergelijkingen op elementen) dat het op een na kleinste element van de heap bepaalt.

2. Wat is de complexiteit van dit algoritme (dus niet de complexiteitsklasse), waarbij we het aantal gemaakte vergelijkingen tellen?

Opgave 3

20 pt

Zij $G = (V, E)$ een gerichte graaf.

1. Geef een DFS algoritme dat bepaalt of G een cykel heeft.
2. Geef de worst-case complexiteit van dit algoritme.

Opgave 4

25 pt

Gegeven een array E met lengte n , dat n verschillende integers bevat.

1. Geef een efficient algoritme dat de lengte van de langste stijgende subreeks van E vindt. Zo'n subreeks hoeft niet opeenvolgende te zijn: bijvoorbeeld, de langste stijgende subreeks van 11,17,5,8,6,4,7,12,3 is 5,6,7,12. Hint: laat $A[i]$ de lengte zijn van de langste stijgende subreeks die begint met $E[i]$, geef een recursieve uitdrukking voor $A[i]$, en bepaal de lengte van de langste stijgende subreeks aan de hand van A . Pas dynamisch programmeren toe.
2. Zorg dat de algoritme ook bepaalt wat de langste stijgende subreeks is.

Opgave 5

10 pt

Geef van de volgende beweringen aan of ze waar of onwaar zijn, en motiveer je antwoord.

1. Beschouw de recurrente betrekking $T(n) = T(\frac{n}{2}) + 2 \cdot \log n$, $T(1) = 1$. Volgens Masters theorema geldt $T(n) \in \Theta(\log n)$.
2. Stel je gebruikt voor open adressering een hastabel met 400 locaties. Stel dat de kans, dat een item op een locatie wordt afgebeeld, voor alle items en alle locaties even groot is. De kans dat het vijfde toegevoegde item to een hashcollision leidt is 0,01.
3. Het Euler circuit probleem (heeft een graaf een cykel waarbij elke edge precies 1 keer bezocht wordt, terwijl knopen vaker mogen voorkomen) is polynomiaal reduceerbaar tot het Hamilton cykel probleem.
4. Een NP-volledig probleem is alleen efficient oplosbaar als het ook in P zit.