

# Tentamen Algoritmen en Datastructuren

dinsdag 18 april 2006, 14 - 17 uur

Het tentamencijfer  $T$  is  $(p/10) + 1$ , waarbij  $p$  het totaal aantal behaalde punten is. Het eindcijfer van het vak is  $(3T + P)/4$ , waarbij  $P$  het practicumresultaat is.

**NB. Beargumenteer je antwoorden, en licht de werking van je algoritmen toe.**

1. (30 punt) Deze opgave gaat over het sorteren van lijsten gehele getallen. Vergelijken van twee getallen gaat in  $O(1)$  tijd.  $n$  is het aantal getallen in de lijst. Alle getallen in de lijst zijn verschillend.
  - (a) Beschrijf het algoritme merge-sort in pseudocode. Laat zien dat de tijdscomplexiteit  $O(n \log n)$  is.
  - (b) Laat zien dat de tijdscomplexiteit van een sorteeralgoritme dat op het vergelijken van getallen gebaseerd is, nooit beter kan zijn dan  $O(n \log n)$ .
  - (c) Nu is tevens gegeven dat alle getallen van de lijst liggen in het interval  $[0, 1000 * n - 1]$ . Geef een algoritme in pseudocode dat de lijst sorteert in  $O(n)$  tijd. Hoeveel geheugen gebruikt het algoritme?
2. (30 punt) Zij gegeven een ongerichte, samenhangende, enkelvoudige (dwz. zonder self-loops en parallelle kanten) gewogen graaf  $G$ , waarin alle gewichten verschillend zijn.
  - (a) Wat is een opspannende boom (spanning tree) van  $G$ ? En een minimale opspannende boom?
  - (b) Zij  $V_1, V_2$  een partitie van de knopen van  $G$ . Beschouw alle kanten van  $G$  die een eindpunt in zowel  $V_1$  als  $V_2$  hebben, en laat  $e$  van die kanten het laagste gewicht hebben. Laat zien: elke minimale opspannende boom van  $G$  bevat  $e$ .
  - (c) Geef een algoritme in pseudocode dat een minimale opspannende boom van  $G$  vindt, met tijdscomplexiteit  $O(m \log n)$  ( $n$  het aantal knopen,  $m$  het aantal kanten). Je mag gebruik maken van een efficiënte priority queue.
3. (30 punt) Deze opgave gaat over complexiteitsklassen.
  - (a) Geef definities van de complexiteitsklassen P (polynomiaal) en NP (nondeterministisch polynomiaal).
  - (b) Wanneer is een probleem NP-volledig (NP-complete)?
  - (c) Formuleer een NP-volledig probleem en laat zien dat het in de klasse NP zit (je hoeft dus niet de NP-volledigheid te bewijzen).