DIT TENTAMEN IS IN ELEKTRONISCHE VORM BESCHIKBAAR GEMAAKT DOOR DE $\mathcal{TBC}$ VAN A–ESKWADRAAT. A–ESKWADRAAT KAN NIET AANSPRAKELIJK WORDEN GESTELD VOOR DE GEVOLGEN VAN EVENTUELE FOUTEN IN DIT TENTAMEN.

Deeltentamen DataStructuren 7 juli 2006
docent Marinus Veldhorst

Schrijf niet met rood en niet met potlood; zet op elk in te leveren vel:
- je naam (met initialen),
- collegekaartnummer,

Besef dat de aangegeven punten bij een opgave niet noodzakelijkerwijs de zwaarte van de opgave aangeven; punten per onderdeel van een opgave kunnen verschillen. Wees zo verstandig om een opgave eerst helemaal door te lezen voordat je een oplossing gaat opschrijven.

1. **(1 punt)**
Zeg van elk van onderstaande beweringen of hij waar (W) of onwaar (N) is. Schrijf je antwoord bij de bewering zelf, en lever het vel bij verlaten van de zaal in.
P.S. k goede antwoorden levert een score van $\frac{\max(0,k-2)}{6}$ punt op.

   (a) <u>Bewering</u>: Voor elke integer $h \geq 0$ is er een propere binaire boom T van hoogte h zodanig dat $depth(u) + height(u) = h$ voor elke knoop $u \in T$.

   (b) <u>Bewering</u>: Voor elke (2,4)-zoekboom T geldt dat in elke interne knoop $u \in T$ òf twee òf vier items zijn opgeslagen.

   (c) <u>Bewering</u>: Voor de functie $f(n) = (\log n)^{100}$ geldt $f(n)$ is $\Omega(\frac{1}{10}n^{1/100})$.

   (d) De postorder nummering van een propere binaire boom T is de toekenning van opeenvolgende nummers (beginnend bij 1) aan knopen in T en wel in de volgorde waarin de knopen bezocht worden door de postorder traversal. Knoop $u \in T$ heeft postorder nummer $p_T(u)$ en analoog inorder nummer $i_T(u)$.
   <u>Bewering</u>: Voor elke propere binaire boom T en voor elke externe knoop $x \in T$ geldt dat $p_T(x) \leq i_T(x)$.

   (e) <u>Bewering</u>: Mergesort werkt in $\Omega(n)$ tijd.

   (f) <u>Bewering</u>: Er is een algoritme dat, gegeven een array A met daarin van klein naar groot n keys, in $O(n)$ tijd een AVL-boom bouwt met daarin de keys van A.

   (g) <u>Bewering</u>: Insertion sort is stabiel en in-place.

   (h) Zij T een functie met als parameter een niet-negatief geheel getal en als uitkomst een niet-negatief geheel getal. Stel $T(0) \leq T(1) \leq T(2) \leq 4$ en voor $n \geq 3$ is $T(n) = 2T(\lfloor n/2 \rfloor) + \lceil 2\sqrt{n} \rceil$.
   <u>Bewering</u>: T is $\Theta(n)$.

2. **Sorteren**
Gegeven $k \geq 2$ sequences $S_1, S_2, \ldots, S_k$ waarvan de elementen integers zijn. Zij n het totaal aantal elementen in de k sequences gezamenlijk.

   (a) **(1.0 punt)** Stel dat alle integers in deze sequences $S_1, S_2, \ldots, S_k$ in het bereik $[0, N-1]$ liggen voor zekere $N \geq 2$. Beschrijf een $O(k + n + N)$-tijd algoritme die elk van de k sequences sorteert.
   Beargumenteer dat je algoritme correct is en voldoet aan de tijdgrens.

   (b) **(1.0 punt)** Stel dat de integers in $S_1, S_2, \ldots, S_k$ willekeurige integers zijn, en dat voor elke j ($1 \leq j \leq k$) de sequence S_j gesorteerd is van klein naar groot. We willen een sequence S maken die alle elementen van $S_1, S_2, \ldots, S_k$ bevat in volgorde van klein naar groot. Geef daarvoor een algoritme dat loopt in tijd $O(n \log k)$.
   Beargumenteer dat je algoritme correct is en voldoet aan de tijdgrens.

   <u>OPMERKING</u>. Je mag in deze opgave algoritmen en/of stellingen uit het boek bekend veronderstellen, maar als je een algoritme/stelling uit het boek gebruikt, moet je expliciet melden dat je dat doet.

3. In deze opgave beschouwen we verzamelingen integers opgeslagen in zoekbomen. Daarmee zijn de verzamelingen van eindige omvang en bevatten ze geen dubbelen. Voor verzamelingen S met minstens twee elementen zijn we geïnteresseerd in de operatie $S.MINDIST$ die twee elementen $s_1, s_2 \in S$ bepaalt zodanig dat $s_1 \neq s_2$ en voor alle $x, y \in S$ $(x \neq y)$ geldt dat $|s_1 - s_2| \leq |x - y|$. $S.MINDIST$ bepaalt daarmee twee verschillende elementen in S met het kleinste onderlinge verschil.

   (a) **(1.5 punt)**
   Stel S is opgeslagen in een binaire zoekboom T. Geef in pseudocode een $O(n)$ tijd algoritme om de $MINDIST$ van S te bepalen. Je algoritme mag naast T niet meer dan $O(h)$ geheugen vragen waarbij h de hoogte van T is.
   Met T is gegeven een pointer naar diens wortel; knopen in T zijn opgeslagen in linked structure met slechts left-pointer, right-pointer, en value-veld.

   In geval we veelvuldig de MINDIST van een verzameling willen weten, is je algoritme aan de trage kant. We overwegen om meer informatie in de knopen van T op te slaan zodat we MINDIST sneller kunnen bepalen, en waarbij inserts en removes (in termen van O) niet langzamer gaan.

   (b) **(1.5 punt)**
   - Geef aan welke informatie je per knoop in T opgeslagen zou willen zien zodanig dat de vier operaties $SEARCH(x)$, $INSERT(x)$, $REMOVE(x)$ en $MINDIST$ elk in $O(h)$ tijd lopen (h hoogte van T). Binnen deze tijdgrens moet de extra informatie ook correct bijgehouden kunnen worden.
   - Geef hiervoor een goed uitgewerkt algoritme voor $S.INSERT(x)$, beargumenteer de correctheid ervan en dat het voldoet aan de tijdgrens.
   - Geef een korte schets van het algoritme voor $S.REMOVE(x)$.

4. **AVL (2 punt)**

   (a) Geef de definitie van een AVL-boom.

   (b) Bewijs de volgende uitspraak:

   > Zij T een AVL-boom met n opgeslagen items en met hoogte h; dan geldt h is $O(\log n)$.

5. **Heaps (2 punt)**

   (a) Geef de definitie van een heap.

   (b) Beschrijf de standaard array-implementatie van een heap.

   (c) Geef een $O(n)$ tijd algoritme dat een array met n integers omvormt naar een heap (in de standaard representatie[1]). Beargumenteer de correctheid van je algoritme, en dat het voldoet aan de tijdgrens.

   (d) Geef minstens 8 tussenstanden van je algoritme bij (c) voor de array

$$(7, 9, 11, 13, 15, 17, 19, 18, 16, 14, 12, 10, 8, 2, 4, 6)$$

——————— *einde toets* ———————

[1] Een uur na aanvang van het tentamen is gemeld dat hier wordt bedoeld: de standaard array-implementatie

2