Kenmerk: EWI2016/TW/DMMP/001/MU_NL (see page 4 for the English version)

Tentamen 1, Module 7, Vakcode 201400433
Discrete Structuren & Efficiënte Algoritmes
Vrijdag 11 maart 2016, 13:45 - 16:45

Alle antwoorden dienen te worden gemotiveerd. Gebruik van een rekenmachine is niet toegestaan. Gebruik van zelfgeschreven formulebladen, één A4 per onderdeel, is wel toegestaan.

Dit tentamen bestaat uit drie onderdelen, en is gebaseerd op de volgende, geschatte tijdsbesteding per onderdeel (slechts als indicatie):

Algorithms & Data Structures (ADS)	1h	(30 punten)
Discrete Mathematics (DW)	1h 20 min	(40 punten)
Languages & Machines (L&M)	40 min	(20 punten)

Dus in totaal 30+40+20=90 punten. Incl. de 10 gratis punten zijn het 100 punten. Het tentamencijfer is het totaal aantal punten gedeeld door 10.

Gebruik aub per onderdeel (ADS/DW/L&M) een nieuw vel!

Algorithms & Data Structures

1. (10 punten) Beschouw het volgende algoritme (met * vermenigvuldigen en // integer division, bv. 7 // 2 = 3):

4x

```
def func(n):
    if n<=1:
        return 100

    value=1
    int k=5
    while k>1:
        value = value*func(n//16)
        k=k-1

    return value
```

   (a) Geef een recurrente betrekking van de tijdscomplexiteit van dit algoritme, uitgedrukt in het aantal rekenkundige operaties.

   (b) Wat is de complexiteitsklasse van dit algoritme?

2. (5 punten)

   Geef een algoritme dat het grootste element in een maxheap verwijdert, en oplevert: een heap met de overgebleven elementen. De complexiteit van het algoritme moet $O(log\ n)$ zijn.

3. (5 punten) Geef een algoritme dat voor een niet-lege binary search tree oplevert: een node met de grootste waarde kleiner-of-gelijk aan de maximumwaarde in de boom (en beargumenteer je oplossing). De boom kan duplicaten bevatten!

4. (10 punten) Gegeven een rugzak met een maximale capaciteit G aan gewicht. Gegeven n voorwerpen 1 t/m n waarbij elk voorwerp i een gewicht w_i heeft. Neem aan dat alle gewichten integers zijn. Het doel is de rugzak in te pakken met zoveel mogelijk gewicht aan voorwerpen.

   (a) Stel je hebt op een bepaald moment nog de voorwerpen 1 t/m i, en je hebt nog g gewicht beschikbaar in je rugzak. Definieer $R(i, g)$ als het ongebruikte gewicht van de rugzak dat je overhoudt als je zoveel mogelijk gewicht aan voorwerpen uit 1 t/m i aan de rugzak toevoegt.
   Beargumenteer welk van de volgende recurrente betrekkingen geldt:
   - $R(i, g) = w_i + max\{R(i-1, g), R(i, g-1)\}$
   - $R(i, g) = min\{R(i-1, g), R(i-1, g-w_i)\}$
   - $R(i, g) = w_i + min\{R(i-1, g), R(i, g)\}$
   - $R(i, g) = min\{R(i-1, g), R(i, g-w_i)\}$

   (b) Geef een algoritme om te bepalen wat het maximum gewicht is dat je in de rugzak kunt stoppen. De complexiteit mag niet slechter zijn dan kwadratisch in n.

Discrete Mathematics

5. (5 punten) Voor gegeven getallen $a, b \in \mathbb{Z}$, neem aan dat $as + bt = 4$ en $ax + by = 7$ voor zekere $s, t, x, y \in \mathbb{Z}$. Laat zien dat a en b relatief priem zijn. connected

6. (7 punten) Laat $G = (V, E)$ een enkelvoudige, samenhangende ongerichte graaf zijn, met $|V| = n$ en $|E| = m$. Laat $d := \frac{1}{n}\sum_{v \in V} d(v)$ de gemiddelde graad van de vertices van G zijn. Laat zien dat, als G planair is, dan is $d < 6$.

7. (10 punten)

   (a) Bereken de oplossing van de recurrente betrekking

   $$a_n - 10a_{n-1} + 25a_{n-2} = 16n + 8 \quad (n \geq 2) \quad \text{met } a_0 = 3 \text{ en } a_1 = 12\,.$$

   (b) Noem a_n het aantal strings uit $\{0, 1, 2\}^*$ van lengte n die geen even aantal 0-en bevatten. Bepaal a_1, a_2, en een recurrente betrekking voor a_n, $n \geq 3$. (Je hoeft deze betrekking niet op te lossen.)

8. (10 punten) Laat $G = (V, E)$ een enkelvoudige, ongerichte graaf zijn, met lijn lengtes $d_e \geq 0$, $e \in E$. Laat $T \subseteq E$ een minimaal opspannende boom (MST) voor G zijn.

   (a) Laat zien dat, als $d_e \neq d_{e'}$ for elke twee lijnen $e, e' \in E$, dan is T uniek.

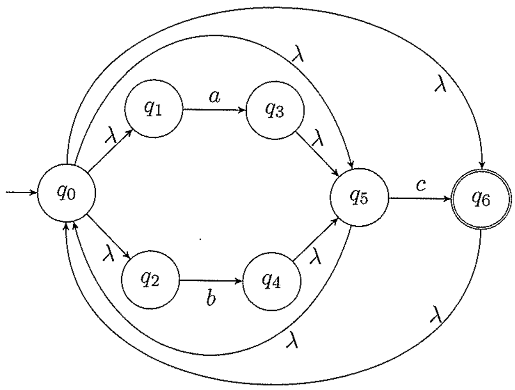
   (b) Laat zien dat het tegenovergestelde niet waar is: Als G een unieke MST T heeft, dan hoeft niet te gelden dat $d_e \neq d_{e'}$ for elke twee lijnen $e, e' \in E$.

   (c) Geef een graaf $G = (V, E)$ met $|V| = n$, met precies n verschillende MST's.

9. (8 punten) Stel twee kratten van verschillende merken bier, á 24 flessen, moeten over 5 controllers verdeelt worden, zodat iedere controller minstens 2 flessen van de éne, en 3 flessen van de andere merk krijgt. Hoeveel mogelijkheden bestaan er voor zo'n verdeling? Gebruik een generende functie.

Languages & Machines

10. (11 punten) Beschouw de volgende NFA-λ, M (alleen q_6 is accepterend):



(a) M is systematisch geconstrueerd uit reguliere expressie E (er missen een paar overbodige λ-stappen). Wat was de originele expressie E?

(b) Geef de λ-closure en de input-transitie functie van M in een tabel.

(c) Transformeer automaat M systematisch naar een DFA.

(d) Geef de minimale DFA en RE voor dezelfde taal. Het antwoord is voldoende, u hoeft deze niet te construeren.

11. (9 punten) We introduceren de volgende 4 talen:

- de taal $L_1 := \{a^i\, b^j\, a^i \mid 0 \leq i \text{ en } 0 \leq j\}$
- de taal $L_2 := \{a^i\, a^i\, b^j \mid 0 \leq i \text{ en } 0 \leq j\}$
- L_3 is een (willekeurige) *eindige* taal.
- L_4 is een (willekeurige) *niet-reguliere* taal.

Geef aan of de volgende talen regulier zijn of niet. Toon je antwoord aan door een constructie of een bewijs te geven.

(a) taal L_1

(b) taal L_2

(c) taal $L_3 \cup L_4$

Kenmerk: EWI2016/TW/DMMP/001/MU_E

Exam 1, Module 7, Code 201400433

Discrete Structures & Efficient Algorithms

Friday, March 11, 2016, 13:45 - 16:45

All answers need to be motivated. No calculators. You are allowed to use a handwritten cheat sheet (A4) per topic (ADS, DM, L&M).

This exam consists of three parts, with the following (estimated):

Algorithms & Data Structures (ADS)	1h	(30 points)
Discrete Mathematics (DW)	1h 20 min	(40 points)
Languages & Machines (L&M)	40 min	(20 points)

Total of 30+40+20=90 points. Including 10 bonus points that makes 100 points. Your exam grade is the total number pf points divided by 10.

Please use a new sheet of paper for each part (ADS/DW/L&M)!

Algorithms & Data Structures

1. (10 points) Consider this algorithm (with * multiplication and // integer division, e.g. 7 // 2 = 3):

```
def func(n):
    if n<=1:
        return 100

    value=1
    int k=5
    while k>1:
        value = value*func(n//16)
        k=k-1

    return value
```

   (a) Give a recurrence relation for the time complexity of this algorithm, expressed in the number of arithmetical operations.

   (b) What is the complexity class of this algorithm?

2. (5 points) Give an algorithm that deletes the biggest element in a maxheap, returning a heap with the remaining elements. The time complexity of this algorithm must be $O(log\ n)$.

3. (5 points) Give an algorithm that yields for a non-empty binary search tree: a node with the biggest value smaller-or-equal than the maximum value in the tree (and explain your solution). The tree may contain duplicates!

4. (10 points) Given a knapsack with maximum weight capacity G. Given n objects 1 to n where each object i has weight w_i. Suppose all weights are integers. The goal is to fill the knapsack with objects, with as much weight as possible.

   (a) Suppose at a certain point you are considering objects 1 to i, and you still have weight g available in the knapsack. Define $R(i, g)$ as the remaining (unused) weight of the knapsack if you have packed as much weight as possible adding objects from 1 to i.
   Motivate which of the following recurrence relations holds:
   - $R(i, g) = w_i + max\{R(i-1, g), R(i, g-1)\}$
   - $R(i, g) = min\{R(i-1, g), R(i-1, g-w_i)\}$
   - $R(i, g) = w_i + min\{R(i-1, g), R(i, g)\}$
   - $R(i, g) = min\{R(i-1, g), R(i, g-w_i)\}$

   (b) Give an algorithm to determine the maximum weight that can be put into the knapsack. The time complexity may be no worse that quadratic in n.

Discrete Mathematics

5. (5 points) For given integer numbers $a, b \in \mathbb{Z}$, assume that $as + bt = 4$ en $ax + by = 7$ for certain $s, t, x, y \in \mathbb{Z}$. Show that a and b are coprime.

6. (7 points) Let $G = (V, E)$ simple, connected, undirected graph, with $|V| = n$ and $|E| = m$. Let $d := \frac{1}{n} \sum_{v \in V} d(v)$ be the average degree of the vertices of G. Show that, if G is planar, then $d < 6$.

7. (10 points)

   (a) Compute the solution to the recurrence relation
   $$a_n - 10a_{n-1} + 25a_{n-2} = 16n + 8 \quad (n \geq 2) \quad \text{with } a_0 = 3 \text{ and } a_1 = 12 .$$

   (b) Let us denote by a_n the number of strings in $\{0, 1, 2\}^*$ of length n that do not contain an even number of 0s. Compute a_1, a_2, and give a recurrence relation for a_n, $n \geq 3$. (You do not have to solve the recurrence relation.)

8. (10 points) Let $G = (V, E)$ be a simple, undirected graph with edge weights $d_e \geq 0$, $e \in E$. Let $T \subseteq E$ be a minimum spanning tree (MST) for G.

   (a) Show that, if $d_e \neq d_{e'}$ for any two edges $e, e' \in E$, then T is unique.

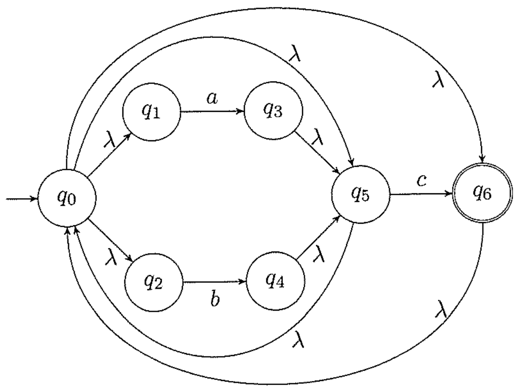
   (b) Show that the opposite is false: If G has a unique MST T, then it need not be true that $d_e \neq d_{e'}$ for any two edges $e, e' \in E$.

   (c) Give a graph $G = (V, E)$ with $|V| = n$, with exactly n different MST's.

9. (8 points) Say we are given two boxes of beer of different brands, each with 24 bottles. The bottles must be distributed over 5 controllers, so that each controller gets at least two bottles of one, and three bottles of the other brand. How many ways are there to distribute the bottles over the 5 controllers? Use a generating function to compute the answer.

Languages & Machines

10. (11 points) Consider the following NFA-λ, M (only q_6 is accepting):



(a) M was constructed systematically from regular expression E (some λ-steps are omitted). What was the original expression E?

(b) Give a table with the λ-closure and input-transition function of M.

(c) Transform the automaton M systematically to an (incomplete) DFA.

(d) Give the minimal DFA and RE for the same language. Just provide the answers, you don't need to provide a construction.

11. (9 points) We introduce the following 4 languages:

- language $L_1 := \{a^i\, b^j\, a^i \mid 0 \leq i \text{ and } 0 \leq j\}$
- language $L_2 := \{a^i\, a^i\, b^j \mid 0 \leq i \text{ and } 0 \leq j\}$
- L_3 is an (abritrary) *finite* language.
- L_4 is an (arbitrary) *non-regular* language.

Indicate whether the following languages are regular or not. Demonstrate your answers by providing a construction or a proof.

(a) language L_1

(b) language L_2

(c) language $L_3 \cup L_4$