TECHNISCHE UNIVERSITEIT DELFT
Faculteit Elektrotechniek, Wiskunde en Informatica

TU Delft

Tentamen IN2505-A Algoritmiek

6 april 2010, 14.00-17.00

- Het gebruik van boek, aantekeningen of rekenmachine tijdens dit tentamen is ***niet*** toegestaan.
- Dit tentamen heeft 15 meerkeuzevragen in totaal goed voor 3 punten en 4 open vragen in totaal goed voor 6 punten. Je krijgt sowieso 1 punt kado. Merk op dat je eindcijfer voor dit vak ook voor $\frac{1}{3}$ uit het onafgeronde gemiddelde van het practicum bestaat (mits beide ≥ 5.0).
- Wat betreft de meerkeuzevragen:
  - Er is voor iedere vraag telkens maar één goed antwoord mogelijk.
  - Alle vragen tellen even zwaar, maar bij de puntentoekenning wordt wel gokkanscorrectie toegepast. Als je geen enkel antwoord geeft, wordt dit altijd fout gerekend.
  - Schrijf je antwoorden eerst op kladpapier en neem ze later over op het antwoordformulier.
  - Vul je studienummer zowel met cijfers in als met streepjes/blokjes.
  - Vul het antwoordformulier bij voorkeur in met pen (geen rode pen) of potlood (goed zwart maken); gebruik geen doorhalingen.
  - Probeer in totaal niet meer dan een uur aan de meerkeuzevragen te besteden. De overige twee uur heb je nodig voor de open vragen.
- De open vragen worden als volgt gewogen:

Vraag:	1	2	3	4	Totaal:
Punten:	6	8	8	8	30

  - Geef antwoord in correct Nederlands of Engels en schrijf leesbaar (gebruik eerst kladpapier).
  - Geef geen irrelevante informatie. Dit kan leiden tot puntenaftrek. Het aangegeven maximaal aantal regels is van toepassing op (getypte) regels en wordt dus aangepast aan de grootte van je handschrift en het gebruik van witruimte.
  - Als er wordt gevraagd om een efficiënt algoritme, geef dan het meest efficiënte algoritme dat je kunt bedenken. Een langzamer algoritme kan minder punten opleveren.
  - Als er wordt gevraagd om pseudocode, dan mag je daarin algoritmen uit het boek aanroepen, tenzij anders vermeld.
  - Voordat je je antwoorden inlevert, controleer of op ieder blaadje je naam en studienummer staat en geef het aantal ingeleverde bladen voor de open vragen aan op (tenminste) de eerste pagina.
- De tentamenstof bestaat uit Chapters 1 to 7 uit Algorithm Design van Kleinberg en Tardos (behalve secties 4.9*, 5.6, 6.10*, 7.4*, en 7.13*), en bovendien de notities over de Master Method en over bewijstechnieken (Proving guide).
- Totaal aantal pagina's: 6.

Open vragen

1. Geef een asymptotische boven- en ondergrens voor $T(n)$ in ieder van de volgende recurrente betrekkingen. Maak je grenzen zo krap (tight) mogelijk en geef aan hoe je aan je antwoord komt (in maximaal 5 regels per recurrente betrekking).
   (a) (3 punten) $T(n) = 4T(n/3) + \log n$
   (b) (3 punten) $T(n) = 3T(n/2) + n^2 \log n$

2. Autofabriek Auto-op-Maat gaat vanwege de kredietcrisis enkele weken dicht. Vanwege een overschot aan personeel lopen de kosten te hoog op. Voordat de fabriek dicht gaat moeten er nog n auto's gemaakt worden. Het in elkaar zetten gebeurt grotendeels door één machine (dus na elkaar), maar daarna moet er nog heel wat maatwerk met de hand gedaan worden. Als er genoeg personeel is (en dat is er), kan het handwerk aan verschillende auto's parallel gebeuren. Voor iedere auto $i \in \{1, \ldots, n\}$ zijn de benodigde tijden bekend: die ene machine heeft eerst m_i uur nodig en daarna is er nog h_i uur handwerk vereist. Gebruik s_i als notatie voor de starttijd waarop de machine aan auto i begint. In deze opgave wordt je gevraagd de optimale starttijden s_i voor iedere auto $i \in \{1, \ldots, n\}$ op de machine te bepalen zodat alle n auto's zo snel mogelijk af zijn en de fabriek (tijdelijk) dicht kan, waarbij $s_i = 0$ voor de eerste auto die door de machine wordt behandeld.

   Beschouw het voorbeeld van drie auto's (dus $n = 3$) in de onderstaande tabel. Het rooster waarbij de machine de auto's produceert in volgorde van hun nummer ($s_1 = 0$, $s_2 = 2$ en $s_3 = 7$) leidt tot een eindtijd van 6 voor auto 1, 17 voor auto 2 en 25 voor auto 3. Dus na 25 uur zijn alle auto's klaar en kan de fabriek gesloten worden. Dit is niet de meest efficiënte volgorde. De beste volgorde is 2, 3, 1 ($s_2 = 0$, $s_3 = 5$ en $s_1 = 16$): de eindtijden zijn dan 15, 23 en 22.

auto i :	m_i	h_i
1	2	4
2	5	10
3	11	7

   (a) (1 punt) Geef een eenvoudige formule voor de tijd f_i dat een auto i helemaal klaar is (zowel met de machine als met het handwerk), afhankelijk van de starttijd s_i en de bovenstaande gegevens, en beschrijf dan de uitdrukking die geoptimaliseerd moet worden (gebruik f_i hiervoor). Geef ook aan of je die uitdrukking wil maximaliseren of minimaliseren (in maximaal 3 regels).
   (b) (3 punten) Geef de pseudocode voor een efficiënt algoritme dat de starttijd s_i voor iedere auto i in een array S invult zodat de fabriek zo snel mogelijk dicht kan (in maximaal 8 regels). (Je mag hierbij bekende algoritmen aanroepen.)
   (c) (1 punt) Geef een krappe bovengrens op de looptijd van je algoritme.
   (d) (3 punten) Gegeven zijn de optimale starttijden o_i voor $1 \leq i \leq n$. Bewijs dat de starttijden s_i van je algoritme uit de voorgaande vraag nooit tot een latere sluiting van de fabriek leiden met behulp van een exchange argument (in maximaal 15 regels). Geef duidelijk aan welke bewijstechniek(en) je gebruikt. Ga er voor het gemak van uit dat alle gegeven tijden uniek zijn.

3. Deze opgave gaat over het benaderen van een verzameling punten met lijnstukken. Gegeven een verzameling punten $(x_1, y_1), \ldots, (x_n, y_n)$, voor iedere $1 \leq i \leq j \leq n$, de kosten $e_{i,j}$ voor de fout die wordt verkregen als de punten i tot en met j worden benaderd met één lijnstuk en de vaste kosten per lijnstuk, C.
   (a) (3 punten) Geef een *recursieve* formule voor de minimale kosten van het benaderen van alle punten met behulp van een verzameling lijnstukken. Je mag hierbij gebruik mag maken van $e_{i,j}$ en C (in maximaal 3 regels). (Denk ook aan de basis van de recursie.)
   (b) (2 punten) Geef pseudocode van een *iteratief* algoritme met behulp van dynamisch programmeren met een looptijd polynomiaal in n dat de minimale kosten van het benaderen van alle punten met behulp van een verzameling lijnstukken berekent (in maximaal 10 regels). Ga ervan uit dat een matrix E met op positie $E[i, j]$ de kosten voor het benaderen van punten i t/m j met één lijnstuk al berekend is.

(c) (1 punt) Geef aan wat de looptijd is van je algoritme en hoe je hieraan komt (in maximaal 3 regels). Neem hiermee de berekening van E niet in mee.

(d) (2 punten) Geef pseudocode voor een algoritme dat (na het uitvoeren van het vorige algoritme) de indices print vanaf waar een nieuw volgend lijnstuk gebruikt wordt (in maximaal 8 regels).

4. ESA probeert bij de volgende ruimtevlucht de netto winst voor commerciële experimenten te maximaliseren en moet daarvoor bepalen welke experimenten gedaan gaan worden en welke instrumenten daarvoor aan boord moeten zijn. Gegeven zijn experimenten $J = \{1, 2, \ldots, m\}$ met opbrengsten p_j voor iedere $j \in J$ en instrumenten $I = \{1, 2, \ldots, n\}$ met kosten c_i voor iedere $i \in I$. Voor ieder experiment j moet een deel $I_j \subseteq I$ van de instrumenten aan boord aanwezig zijn.

(a) (3 punten) Modelleer dit probleem door middel van een stroomnetwerk (flow network) zodat uit de minimale snede (A, B) afgeleid kan worden welke experimenten gedaan moeten worden en welke instrumenten meegenomen moeten worden om de netto winst te maximaliseren. Beschrijf wat de knopen zijn, wat de kanten zijn en wat de capaciteiten op de kanten zijn (in maximaal 8 regels). Maak ook een schets van een eenvoudig voorbeeld.

(b) (3 punten) Leg uit hoe uit een (minimale) snede (A, B) de geselecteerde experimenten en instrumenten afgeleid kunnen worden en bewijs dat altijd alle instrumenten benodigd voor de geselecteerde experimenten aanwezig zijn (in maximaal 8 regels).

(c) (2 punten) Leg uit hoe de netto winst berekend kan worden uit een (minimale) snede (A, B). Bewijs dat de netto winst het grootst is bij de minimale snede (in maximaal 8 regels).

Einde tentamenopgaven