DATALOGISK INSTITUT, AARHUS UNIVERSITET

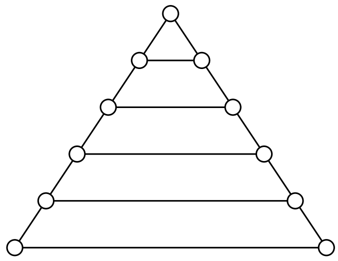
Det Naturvidenskabelige Fakultet
EKSAMEN
Grundkurser i Datalogi
Algoritmer og Datastrukturer 2 (2003-ordning)
Antal sider i opgavesættet (incl. forsiden): 5 (fem)
Eksamensdag: Mandag den 14. august 2006, kl. 9.00-13.00
Eksamenslokale: Trøjborg, Willemoesgade 15, Århus N, 8200 Århus N
Tilladte medbragte hjælpemidler: Alle sædvanlige hjælpemidler (lærebøger og notater)
Materiale der udleveres til eksaminanden:

OPGAVETEKSTEN
BEGYNDER
PÅ NÆSTE SIDE

—oOo—

Opgave 1 (25%)

I denne opgave betragtes *pyramide-grafer*. En pyramide-graf med k lag består af en trekant øverst med yderligere $k-1$ lag herunder, som hver tilføjer yderligere to knuder til grafen. Nedenstående figur viser en pyramide-graf med 5 lag.



Spørgsmål a: Angiv antal knuder n og kanter m i en pyramide-graf som funktion af antal lag k. Angiv som funktion af k udførselstiden for Prim's algoritme for at finde et minimum udspændende træ af en vægtet pyramide-graf. □

Man kan i det følgende bruge nedenstående sætning om minimum udspændende træer uden bevis:

> *Lad G være en vægtet uorienteret graf hvor alle kanter har forskellige vægte. Lad e være en kant i G. Et minimum udspændende træ for G indeholder e hvis og kun hvis G ikke indeholder en cykel C, hvor e er den tungeste kant i cyklen C.*

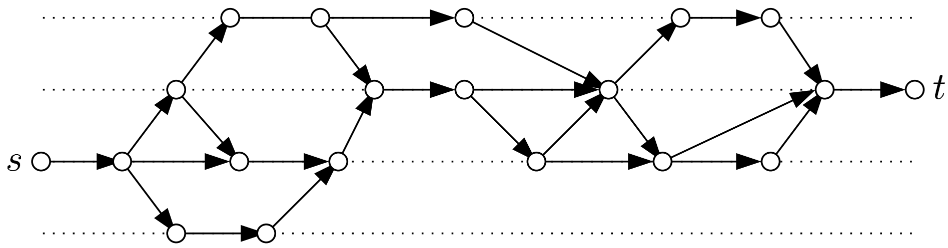
Spørgsmål b: Beskriv en algoritme der finder et minimum udspændende træ for en pyramide-graf med n knuder i tid $O(n)$. Det kan antages at alle vægte er forskellige. □

Spørgsmål c: Beskriv en algoritme med udførselstid $O(n)$ der finder den længste sti mellem to knuder i et minimum udspændende træ for en pyramide-graf med n knuder. □

(Opgavesættet fortsætter)

Opgave 2 (25%)

I det følgende betragtes *k-spors grafer*, som er orienterede grafer hvor knuderne er placeret på k horizontale linier/spor, og hvor alle knuder højest har tre udgående kanter: en kant til den næste knude til højre i samme spor, en kant til en knude til højre i sporet ovenover, og en kant til en knude til højre i sporet nedenunder.



Spørgsmål a: Giv en algoritme med udførselstid $O(n)$, der givet en vægtet k-spors graf med n knuder og to knuder s og t, finder den kortest vej fra s til t. □

Spørgsmål b: Giv en algoritme med udførselstid $O(n)$, der givet en k-spors graf med n knuder og to knuder s og t, finder en vej fra s til t med mindst mulige spor-skifte, dvs. bruger færrest mulig op- og nedkanter. □

Spørgsmål c: Giv en algoritme, der givet en vægtet k-spors graf med n knuder og to knuder s og t, finder den kortest vej fra s til t hvor alle op og ned kanter på stien er efterfulgt af en horizontal kant (dvs. det er ikke tilladt at have to kanter umiddelbart efter hinanden som begge skifter mellem to spor). Angiv algoritmens udførselstid. □

Opgave 3 (25%)

I denne opgave er der givet to sekvenser af heltal $x_1, x_2, \ldots, x_n$ og $y_1, y_2, \ldots, y_m$ med henholdsvis n og m elementer. I denne opgave ønsker vi at finde en voksende delsekvens der alternerer mellem elementer fra de to lister, og som starter med et element x_i og slutter med et element y_j. Formelt ønsker vi at finde $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ og $1 \leq j_1 < j_2 < \cdots < j_k \leq m$ således at k er størst mulig og $x_i < y_i$ for $i = 1, 2, \ldots, k$ og $y_i < x_{i+1}$ for $i = 1, 2, \ldots, k-1$. Vi kalder sådan en sekvens for en *længste voksende alternerende delsekvens.*

For de to sekvenser

$$
\begin{aligned}
x_1, \ldots, x_n &= \underline{3}, \underline{7}, 15, 3, 16, 4, \underline{12}, 8 \\
y_1, \ldots, y_m &= 4, 9, 13, 10, \underline{5}, 6, \underline{8}, \underline{14}, 15
\end{aligned}
$$

er en længste voksende alternerende delsekvens $3, 5, 7, 8, 12, 14$.

I det følgende er $L(i,j)$ længden af en længste voksende alternerende delsekvens der slutter med x_i og y_j. $L(i,j)$ kan beskrives ved følgende rekursionsformel:

$$
L(i,j) = \begin{cases} 0 & \text{hvis} \quad i = 0 \vee j = 0 \\ 0 & \text{hvis} \quad x_i \geq y_j \\ 2 + \max\{L(i',j') \mid 0 \leq i' < i \wedge 0 \leq j' < j \wedge y_{j'} < x_i\} & \text{hvis} \quad x_i < y_j \end{cases}
$$

Spørgsmål a: Udfyld nedenstående $L(i,j)$ tabel for sekvenserne $x_1, \ldots, x_4 = 3, 5, 1, 4$ og $y_1, \ldots, y_5 = 6, 2, 7, 8, 1$:

$i \backslash j$	0	1	2	3	4	5
0						
1						
2						
3						
4						

□

Spørgsmål b: Angiv en algorime baseret på dynamisk programmering der finder længden af en længste voksende alternerende delsekvens for $x_1, \ldots, x_n$ og $y_1, \ldots, y_m$. Angiv algoritmens udførselstid. □

Spørgsmål c: Udvid algoritmen til at rapportere en længste voksende alternerende delsekvens. Angiv algoritmens udførselstid. □

Opgave 4 (25%)

I denne opgave betragtes kun strenge over alfabetet $\{a, b\}$.

For en streng $T = T[1]T[2]\cdots T[m]$ af længe m definerer vi *perioden* af T til at være det mindste $p > 0$ således at

$$T[p+1..m] = T[1..m-p] \ ,$$

dvs. $T[p+i] = T[i]$ for $i = 1, 2, \ldots, m-p$.

Strengen $T =$ `abaabaab` har perioden 3 da $T[4..8] = T[1..5] =$ `abaab`, hvilket også kan ses ved at placere to kopier af T over hinanden med den ene forskudt p positioner:

$$\begin{array}{ccccccccccc}
\multicolumn{3}{c}{\underbrace{\mathtt{a\ b\ a}}_{p}} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{b} & & & \\
 & & & | & | & | & | & | & & & \\
 & & & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{b}
\end{array}$$

En streng T af længde m siges at være *periodisk* hvis dens periode $p \leq m/2$.

Ovenstående streng er periodisk da $p = 3 \leq m/2 = 4$. Derimod er strengen `ababbbab` af længde $m = 8$ ikke periodisk, da den har perioden $p = 6 > m/2 = 4$.

Spørgsmål a: Angiv perioden for strengen $T =$ `abbabbaabbabb` og om strengen er periodisk. □

Spørgsmål b: Beskriv en algoritme med udførselstid $O(m^2)$ der beregner perioden af en streng T af længde m. □

I det følgende ønsker vi givet en streng S af længde n at finde en *længste periodiske delstreng* af S (hvis sådan en delstreng findes). Den længste periodiske delstreng af $S = \mathtt{b\underline{babaababaab}bbabbaa}$ er delstrengen $S[2..12]$ som har perioden 5.

Bemærk at i suffix-træet for ovenstående streng S, findes der en knude v svarende til delstrengen `babaab`, v har i sit undertræ blade svarende til suffixerne af S starende i position 2 og 7, perioden $p = 7 - 2$, og at $|\mathtt{babaab}| = 6 \geq p = 5$.

Spørgsmål c: Beskriv en algoritme med udførselstid $O(n^2)$, der givet et suffix-træ for en streng S af længde n, for hver knude v i suffix-træet finder to positioner i_v og j_v hvor suffixerne startende i position i_v og j_v har delstrengen svarende til v som prefix, og således at afstanden mellem i_v og j_v er mindst mulig. □

I det følgende kan antages at et suffix-træ for en streng af længde n fra et alfabet med $O(1)$ tegn kan konstrueres i $O(n)$ tid.

Spørgsmål d: Angiv en algoritme der givet en streng S af længde n finder den længste periodiske delstreng i tid $O(n^2)$. □