

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Nå også som prøveeksamen 2007

Forsøk deg selv på tre timer! Gjennomgås 29/11-2007, 14.15, Lille Aud

Eksamen i:	INF 3130/4130: Algoritmer: Design og effektivitet
Eksamensdag:	Fredag 15. desember 2006
Tid for eksamen:	Kl. 09.00 til 12.00
Oppgavesettet er på:	4 sider
Vedlegg:	Ingen
Tillatte hjelpemidler:	Alle trykte og skrevne

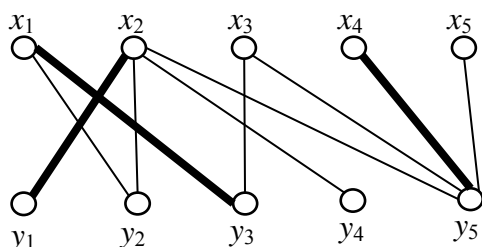
Kontroller at oppgavesettet er komplett før du begynner å besvare spørsmålene.

Les oppgavene nøye, og lykke til!

Oppgave 1.

Spørsmål 1a (7%)

Under er gitt en bipartit graf, med en angitt (ikke-perfekt) matching. Finn (gjerne på en intuitiv måte) en forbedringsvei i forhold til denne matchingen. Angi denne forbedringsveien og tegn grafen med den matching du får om du “bruker” denne forbedringsveien.

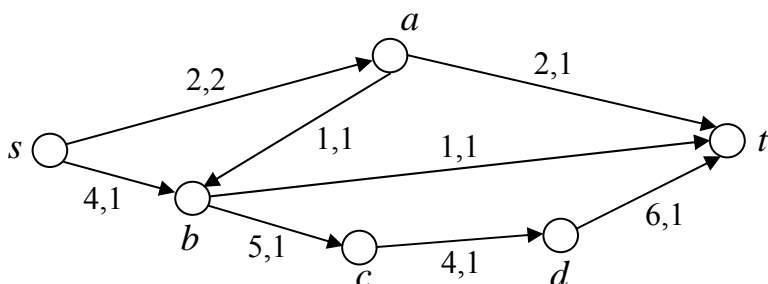


Spørsmål 1b (7%)

Påvis at den underliggende grafen fra 1a ikke har noen større matching enn den du fant i 1a.

Spørsmål 1c (7%)

Under er gitt et flytnettverk, der målet er å maksimere flyten fra s til t . På hver kant er angitt en kapasitet og en flyt (i den rekkefølgen). Vi tenker oss at vi bruker FordFulkerson-algoritmen, med reglen angitt av Edmonds og Karp for å velge neste steg. Hva blir da det neste steget algoritmen gjør?



Oppgave 2

Hvilke av språkene angitt under er uavgjørbare? Begrunn svaret.

Spørsmål 2a (7%)

$L_1 = \{M \mid M \text{ stopper for enhver input}\}$

Spørsmål 2b (7%)

$L_2 = \{M \mid M \text{ løser Stoppeproblemet}\}$

Oppgave 3

Fire-dimensjonal matching (4DM) er en rett fram generalisering av *Tre-dimensjonal matching*. (Sitat fra foilene:

3-DIMENSIONAL MATCHING (3DM)

Instance: A set M of triples (a, b, c) such that $a \in A, b \in B, c \in C$. All 3 sets have the same size q ($|A| = |B| = |C| = q$).

Question: Is there a matching in M , i.e. a subset $M' \subseteq M$ such that every element of A, B and C is part of exactly 1 triple in M' ?)

Spørsmål 3a (7%)

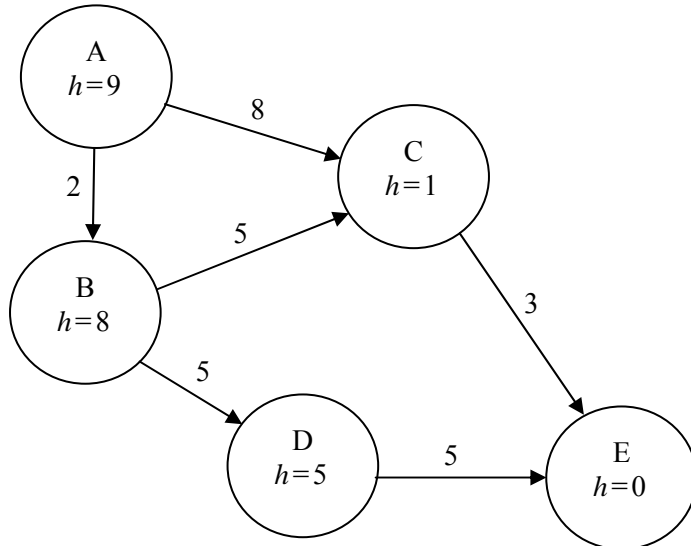
Angi spesifikt hvorfor man kan si at 4DM (og 3DM) er i NP, ved å ta utgangspunkt i definisjonen av NP.

Spørsmål 3b (8%) (Vente med denne til slutt?)

Bevis at 4DM er NP-komplett. (Hint: Ta utgangspunkt i beviset for at 3DM er NP-komplett.)

Oppgave 4

Gitt følgende rettede graf, hvor vi ønsker å bruke A*-søk (altså "A*-SearchMH(...) i boka) til å finne korteste vei fra A til E.



Her er alle veilengder (eller "costs" eller "weights") for enkelt-steg angitt som tall på kantene (i boka er dette c -funksjonen, slik at vi her altså har $c(A, B) = 2$). Dessuten er den heurstikk-funksjonen h som skal brukes slik at den for de forskjellige nodene blir som angitt (når E er målnode).

Spørsmål 4a (7%)

Gjennomfør et søk fra A til E med A*-algoritmen. Angi stegene som algoritmen går gjennom så nøye at du kan påvise at algoritmen faktisk ikke finner korteste vei fra A til E. Angi f- og g-verdier.

(**Merk:** Det er en trykkfeil på foilen som beskriver A*-algoritmen (med foil-overskrift: "Selve algoritmen"). Linja som starter med: "- Ta ny minste ut av PQ, ...", skal erstattes med: "- Ellers, la x bli en tre-node".)

Spørsmål 4b (7%)

Finn grunnen til at A*-algoritmen her ikke finner korteste vei. Angi denne generelt, og eksplisitt i forhold til grafen over.

Oppgave 5

Spørsmål 5a (7%)

Vi skal se på følgende problem: Gitt to bokstavsekvenser S og T. Vi skal avgjøre om man kan lage S ut fra T ved kun å fjerne tegn fra T. Om S = "abc" og T = "aacbbac" er dette mulig, men ikke om T = "aacbba".

Angi hvordan man kan løse dette problemet med dynamisk programmering. Angi her hva slags tabell du vil bruke, hvordan den generelle reglen for utfylling er, og hvordan du vil initialisere tabellen. Skisser også et program som gjør dette.

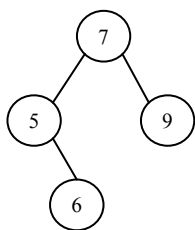
Spørsmål 5b (8%) (Vente med denne til slutt?)

Forklar hvordan du kan forandre din algoritme fra 5a slik at den løser følgende problem: Man kan fremdeles stryke tegn i T, men får ikke lov til å stryke to tegn som står etter hverandre i (den opprinnelige) T. Er det da mulig å få fram S fra T? (Hint: Forsøk å bruke to varianter av “true” i tabellen.)

Oppgave 6

Spørsmål 6a (7%)

Vi starter med det følgende splay-treet:



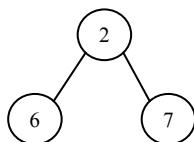
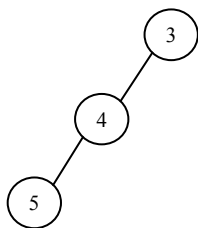
Tegn treet vi ender opp med når noden med verdi 6 slettes. Angi hvilken aksess-sti som følges og hva slags rotasjon som utføres.

Spørsmål 6b (7%)

Vis så resultatet av å sette inn verdien 8 i det **opprinnelige** splay-treet i spørsmål 6a.

Spørsmål 6c (7%)

Vi har følgende to venstreviddede heaper (leftist heaps) L1 og L2:



Vis resultatet av å spleise L1 og L2 ($\text{merge}(L1, L2)$).

(Slutt på oppgavesettet)