Skriftlig Eksamen
Algoritmer og Datastrukturer (DM02)

Institut for Matematik og Datalogi
Syddansk Universitet, Odense

Mandag den 17. januar 2005, kl. 9–13

Alle sædvanlige hjælpemidler (lærebøger, notater, osv.) samt brug af lommeregner er tilladt.

Eksamenssættet består af 5 opgaver på 5 nummererede sider (1–5).
Fuld besvarelse er besvarelse af alle 5 opgaver.
De enkelte opgavers vægt ved bedømmelsen er angivet i procent. Bemærk, at de enkelte spørgsmål i en opgave ikke nødvendigvis har samme vægt.

Der må gerne refereres til algoritmer og resultater fra lærebogen inklusive øvelsesopgaverne. Henvisninger til andre bøger accepteres ikke som besvarelse af et spørgsmål.

Bemærk, at hvis der er et spørgsmål, man ikke kan besvare, må man gerne besvare de efterfølgende spørgsmål og blot antage, at man har en løsning til de foregående spørgsmål.

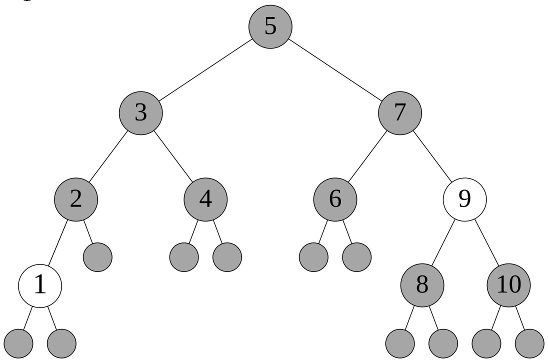
Husk at begrunde dine svar!

Opgave 1 (15%)

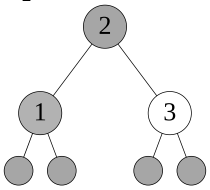
Nedenfor er tegnet nogle binære træer med røde og sorte knuder. Grå cirkler symboliserer sorte knuder, og hvide cirkler symboliserer røde knuder.

Spørgsmål a: Hvilke af følgende fire træer er rød-sorte træer? Begrund dine svar.

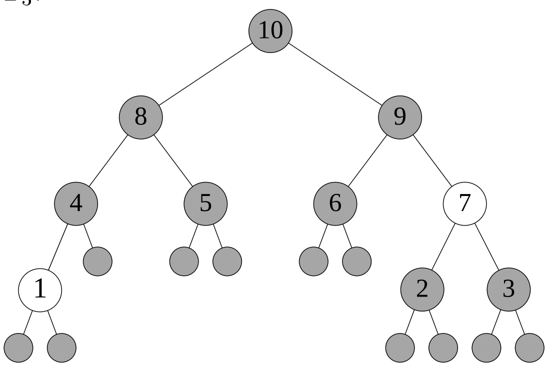
T_1:



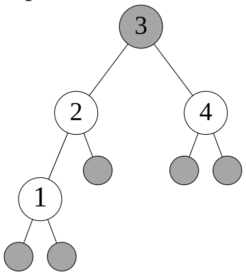
T_2:



T_3:

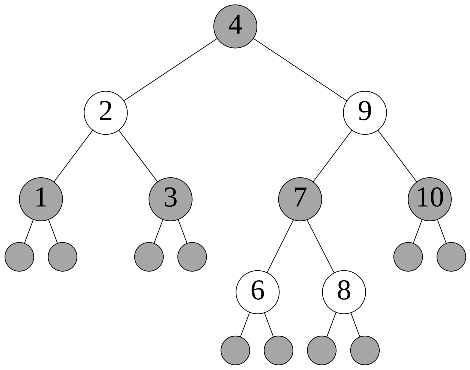


T_4:



□

Spørgsmål b: Betragt følgende rød-sorte træ.



Nu indsættes et element med nøgle 5. Tegn træet, som det ser ud efter indsættelsen. Vis de enkelte trin i indsættelsen. □

Opgave 2 (15%)

Betragt følgende algoritme til at gange to positive heltal.

<u>MULT</u>(a,b)
Hvis $a = 1$
    returner b
Ellers
    returner MULT $\left(\left\lfloor \frac{a}{2} \right\rfloor, 2b\right) + b \cdot (a \bmod 2)$

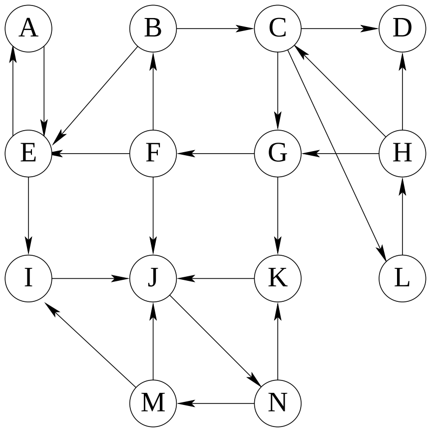
(Husk, at $\lfloor \frac{a}{2} \rfloor$ betyder "$\frac{a}{2}$ rundet ned".)

Spørgsmål a: Bevis ved induktion over a, at MULT(a, b) returnerer produktet ab. □

Spørgsmål b: Lad $T(a)$ udtrykke algoritmens asymptotiske køretid (tidskompleksitet) som funktion af første argument. Opskriv en rekursionsligning for $T(a)$, og løs ligningen. □

Opgave 3 (25%)

Betragt den orienterede graf $G = (V, E)$ nedenfor.



Spørgsmål a: Find de stærke sammenhængskomponenter i G.

Du skal angive præ- og postnumre for det første dybde-først-gennemløb af grafen. Tegn desuden den udspændende skov, som opnås i andet gennemløb.

I de tilfælde, hvor algoritmen giver frit valg m.h.t. hvilken knude, der skal behandles som den næste, **skal** knuderne vælges i alfabetisk rækkefølge. $\square$

Komponentgrafen $G' = (V', E')$ for G er som bekendt en acyklisk graf, som har en knude v_i for hver stærk sammenhængskomponent $C_i = (V_i, E_i)$ i G. For hvert par af knuder $v_i, v_j \in V'$ er der en kant fra v_i til v_j, hvis og kun hvis der i G er en kant fra en knude i C_i til en kant i C_j. D.v.s. $E' = \{(v_i, v_j) \mid \exists x_i \in V_i, x_j \in V_j \colon (x_i, x_j) \in E\}$.

Spørgsmål b: Tegn komponentgrafen G' for G.

Knuderne i G' skal navngives ud fra knudenavnene i G. Hvis en knude i G' f.eks. svarer til en komponent i G bestående af knuderne X, Y og Z, skal knuden navngives XYZ. $\square$

Spørgsmål c: Angiv en topologisk sortering af knuderne i komponentgrafen fra spørgsmål b. Gør rede for, hvorfor det er en topologisk sortering. $\square$

Opgave 4 (25%)

Denne opgave minder om at finde en længste fælles delsekvens (longest common subsequence) for to strenge. Vi er dog interesserede i at finde en sekvens, der ikke blot er så lang som muligt, men også så sammenhængende som muligt.

Mere præcist betragtes to strenge $S = s_1 s_2 \ldots s_m$ og $T = t_1 t_2 \ldots t_n$. Antag, at F er en fælles delsekvens for S og T, og lad b være det mindste antal blokke, F kan deles op i, så hver blok er en sammenhængende delsekvens af både S og T. *Blokscoren* af F er $|F| - b$.

Blokscoren for et par af strenge er blokscoren af den (eller de) delsekvens(er), som har den største blokscore.

F.eks. har nedenstående to strenge blokscore $2 + 4 + 4 - 3 = 7$.

$$S = \boxed{ab}\,cc\,\boxed{baca}\,ba\,\boxed{abad}$$
$$T = \boxed{ab}\,aba\,\boxed{baca}\boxed{abad}\,ad$$

Lad $B(i, j)$ være blokscoren af $S_i = s_1 s_2 \ldots s_i$ og $T_j = t_1 t_2 \ldots t_j$.

Spørgsmål a: Lad $S = bbcba$ og $T = bccbba$. Udfyld nedenstående tabel med B-værdierne for S og T. D.v.s. på plads (i, j) skrives $B(i, j)$. $\square$

		j						
		0	1	2	3	4	5	6
i	0							
	1							
	2							
	3							
	4							
	5							

Spørgsmål b: Gør rede for, at $B(i,j)$ opfylder følgende rekursionsligning.

$$B(i,j) = \begin{cases} 0, & \text{hvis } i = 0 \lor j = 0 \\ \max\left\{B(i,j-1), B(i-1,j)\right\}, & \text{hvis } s_i \neq t_j \\ \max\left\{B(i,j-1), B(i-1,j), \right. & \\ \qquad \left. B(i-m_{ij}, j-m_{ij}) + m_{ij} - 1\right\}, & \text{hvis } s_i = t_j \end{cases}$$

hvor $m_{ij} = \max\{k \mid s_{i-\ell} = t_{j-\ell}, 0 \leq \ell \leq k-1\}$. D.v.s. m_{ij} angiver det største k, så de sidste k tegn i S_i og T_j matcher. □

Spørgsmål c: Skriv i pseudokode en algoritme, som tager to strenge som input og beregner deres blokscore. Algoritmen skal bruge dynamisk programmering.

Hvad er algoritmens asymptotiske køretid? □

Opgave 5 (20%)

Denne opgave handler om at sortere n heltal med mange dubletter.

Spørgsmål a: Beskriv en algoritme, der sorterer n heltal i tid $O(n \log(\log n))$, hvis der kun er $O(\log n)$ forskellige tal. Det antages, at to tal kan sammenlignes i konstant tid (d.v.s. i tid $O(1)$).
Hint: Benyt en passende valgt datastruktur til at indsætte i, og saml dubletterne.

Bemærk, at forskellen mellem to tal sagtens kan være $\omega(\log n)$. D.v.s. selvom der kun er $O(\log n)$ forskellige tal, kan der være stor forskel på størrelsen af tallene.

Forklar, hvordan den sorterede følge kan aflæses fra datastrukturen og udskrives i tid $O(n)$. □