Skriftlig Eksamen
Algoritmer og Datastrukturer (DM507)

Institut for Matematik og Datalogi
Syddansk Universitet, Odense

Tirsdag den 8. januar 2008, kl. 9–13

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

Eksamenssættet består af 4 opgaver på 7 nummererede sider (1–7).
Fuld besvarelse er besvarelse af alle 4 opgaver.
De enkelte opgavers vægt ved bedømmelsen er angivet i procent.

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 (20%)

Spørgsmål a (8%): Betragt alfabetet med de syv tegn a, b, c, d, e, f, g. Nedenstående tabel viser, hvor tit hvert enkelt tegn optræder i en given tekst.

a	b	c	d	e	f	g
9	7	24	10	55	15	25

Tegn et Huffman-træ, som repræsenterer Huffman-koderne for dette eksempel. □

Spørgsmål b (7%): Indsæt nøglen 9 i den binære max-hob repræsenteret ved nedenstående array.

10	8	6	3	7	4	5	1	2	

Vis resultatet før hvert gennemløb af `while`-løkken i algoritmen på s. 140 i lærebogen.

Du må gerne tegne det som en træstruktur i stedet for et array. □

Spørgsmål c (5%): Dette spørgsmål handler om hashing med lineær probing. Betragt følgende hashtabel med 10 pladser

0	1	2	3	4	5	6	7	8	9
	11	31		24	15			48	

og følgende hashfunktion

$$h(k) = k \bmod 10$$

Som det ses, er der allerede indsat fem elementer i hashtabellen.

Nu indsættes et element med nøgle 4. På hvilken plads havner dette element? □

Opgave 2 (25%)

Denne opgave handler om letteste udspændende træer (minimum spanning trees) og korteste veje i vægtede grafer.

Spørgsmål a (9%): Lad G være en vægtet graf, og lad e være en kant i G. Antag, at der findes en kreds C, sådan at e er den tungeste kant i C, dvs.

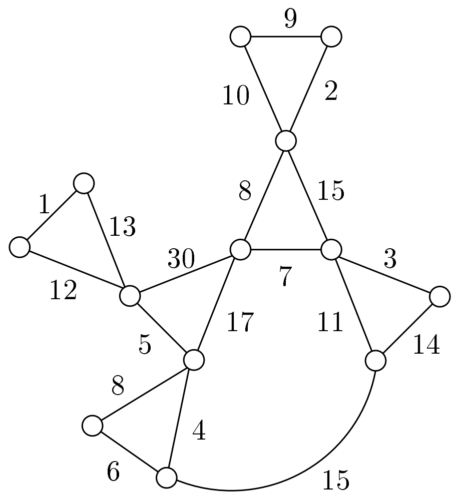
$$e \in C \quad \text{og} \quad w(e) > w(e') \text{ for alle } e' \in C - \{e\}.$$

Bevis, at e ikke kan være med i et letteste udspændende træ for G. $\square$

Spørgsmål b (8%): Tegn et letteste udspændende træ for nedenstående vægtede graf.

Du kan evt. bruge tegningen på sidste side.

Argumenter for, at dit resultat er et letteste udspændende træ.

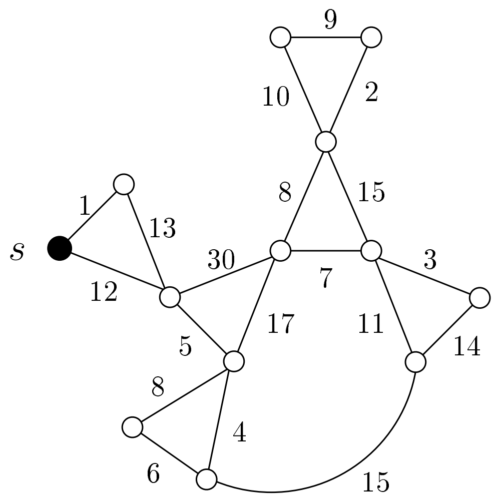


$\square$

Spørgsmål c (8%): Vi ser igen på den samme graf som i spørgsmål b (se nedenfor). Tegn et korteste-vej-træ med rod i den sorte knude s. Dvs. tegn de korteste veje fra s til hver af de andre knuder i grafen.

Du kan evt. bruge tegningen på sidste side.

Husk at argumentere for, at dit resultat er rigtigt.

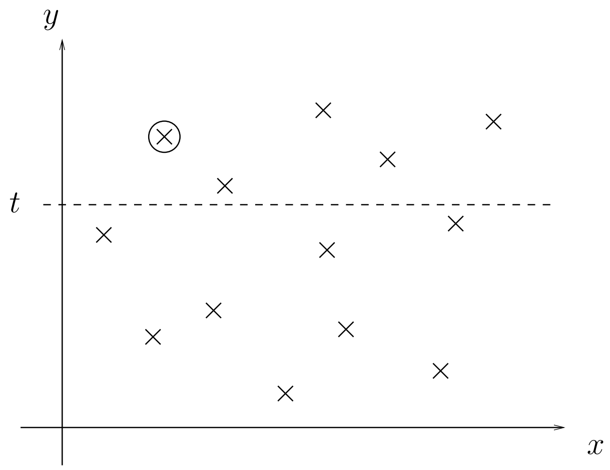


$\square$

Opgave 3 (30%)

Lad $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ være punkter i planen. Antag, at koordinaterne er positive heltal, og at alle x-koordinaterne er forskellige.

Vi skal se på operationen `MinAbove`(t). Hvis der findes mindst ét punkt, hvis y-koordinat er større end eller lig med t, returneres det af disse punkter, som har mindst x-værdi. Ellers meddeles, at et sådant punkt ikke findes. I nedenstående eksempel er det punktet med cirkel omkring, som returneres.



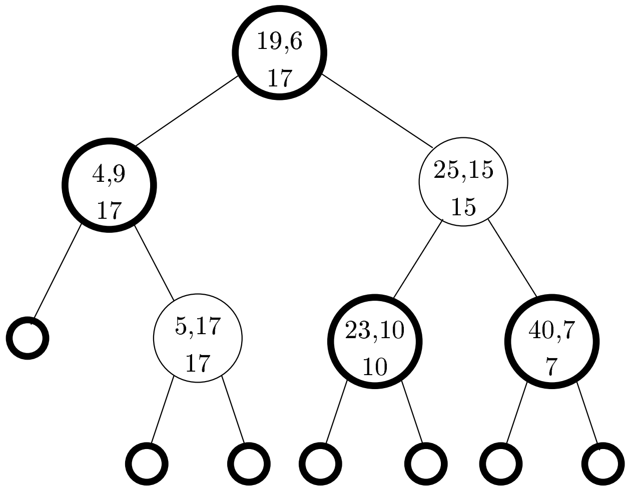
Spørgsmål a (5%): Hvad er `MinAbove`(10) for punkterne

(4,9), (5,17), (19,6), (23,10), (25,15), (40,7)?

□

I det følgende betragtes en udvidelse af rød-sorte træer til opbevaring af punkterne. Hver knude indeholder et punkt, dvs. x-koordinat og y-koordinat, og x-koordinaterne anvendes som nøgler i træet. Desuden indeholder hver knude et tal $y_{\max}$, som angiver den største y-koordinat i knudens undertræ.

Nedenfor er vist et udvidet rød-sort træ for punkterne i spørgsmål a. I knuderne er angivet x, y øverst og $y_{\max}$ nederst. Sorte knuder er tegnet med fed streg.



Spørgsmål b (10%): Beskriv, hvordan et udvidet rød-sort træ kan vedligeholdes under indsættelse og sletning af punkter. □

Spørgsmål c (7%): Illustrer en del af svaret fra spørgsmål b ved at indsætte punktet (30,11) i det udvidede rød-sorte træ, som er vist øverst på denne side. □

Spørgsmål d (8%): Beskriv, hvordan `MinAbove`(t) kan udføres i tid $O(\log n)$. □

Opgave 4 (25%)

I denne opgave er der givet en sorteret sekvens af positive heltal $X = \langle x_1, x_2, \ldots, x_n \rangle$ og et positivt heltal W. Vi ønsker at finde en delsekvens $Y = \langle y_1, y_2, \ldots, y_m \rangle$ af X, hvor $y_1 = x_1$ og $y_m = x_n$. Målet er, at forskellen på y_i og y_{i+1} skal være så tæt på W som muligt, for hvert $i = 1, 2, \ldots, m-1$. Mere præcist defineres *ujævnheden* U af Y ved

$$U = \sum_{i=1}^{m-1} (y_{i+1} - y_i - W)^2,$$

og vi ønsker at finde en delsekvens med mindst mulig ujævnhed.

> **Eksempel:** Lad $X = \langle 2, 4, 8, 9, 11, 13, 14, 20 \rangle$ og $W = 5$.
>
> Delsekvensen $Y_1 = \langle 2, 8, 13, 20 \rangle$ har ujævnhed
>
> $$\begin{aligned} U &= (8-2-5)^2 + (13-8-5)^2 + (20-13-5)^2 \\ &= 1^2 + 0^2 + 2^2 \;=\; 5. \end{aligned}$$
>
> Delsekvensen $Y_2 = \langle 2, 8, 14, 20 \rangle$ har ujævnhed 3, og det er mindst muligt.

For $1 \leq k \leq n$, betegner $U(k)$ den *mindste mulige ujævnhed* af en delsekvens af $\langle x_1, x_2, \ldots, x_k \rangle$. Specielt er $U(n)$ den mindste mulige ujævnhed af en delsekvens af hele sekvensen X. Dvs. for sekvensen X i eksemplet ovenfor er $U(8) = 3$. Bemærk, at $U(1) = 0$ for enhver sekvens X.

Spørgsmål a (10%): Udfyld nedenstående tabel for

$$X = \langle 2, 6, 8, 10, 14, 15, 20, 21 \rangle \text{ og } W = 7.$$

Du kan evt. bruge tabellen på sidste side.

k	1	2	3	4	5	6	7	8
$X[k]$	2	6	8	10	14	15	20	21
$U(k)$								

□

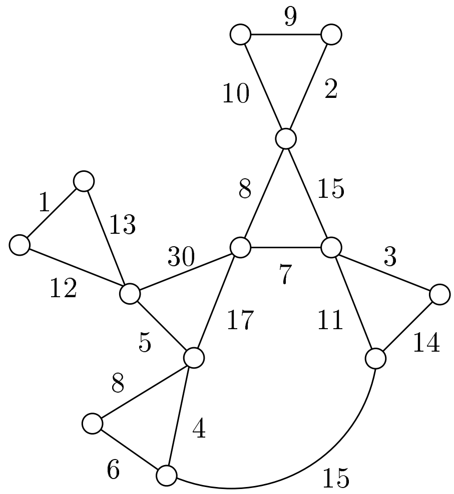
Spørgsmål b (5%): Lad X og W være som i spørgsmål a.

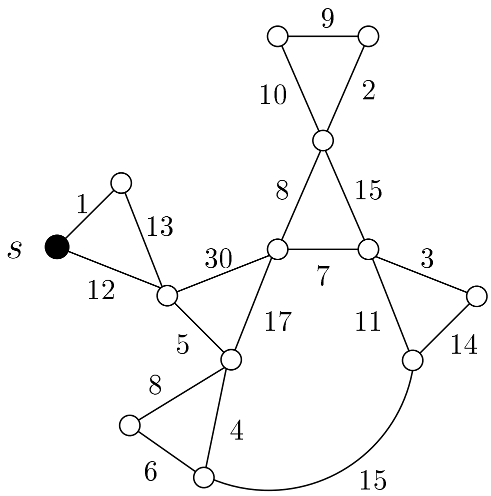
Angiv en delsekvens med mindst mulig ujævnhed. □

Spørgsmål c (10%): Angiv en algoritme baseret på dynamisk programmering, som finder den mindste mulige ujævnhed for en delsekvens af X.

Du behøver ikke at finde den delsekvens, som svarer til den mindste ujævnhed.

Hvad er din algoritmes køretid og pladsforbrug? □





k	1	2	3	4	5	6	7	8
$X[k]$	2	6	8	10	14	15	20	21
$U(k)$								