

DATALOGISK INSTITUT, AARHUS UNIVERSITET

Science and Technology
RE-EKSAMEN
<b>Grundlæggende algoritmer og datastrukturer</b>
Antal sider i opgavesættet (incl. forsiden): 12
Eksamensdag: <b>Mandag d. 28. maj 2018, kl. 9.00 - 11.00</b>
Eksamenslokale: Trøjborg 82-84, Bygning 1912-1917
Tilladte medbragte hjælpemidler: INGEN
Materiale der udleveres til eksaminanden: Opgavesæt

OPGAVETEKSTEN  
BEGYNDER  
PÅ NÆSTE SIDE

---oOo---

## FADS 17 - Re-Eksamen

Dette eksamenssæt består af to dele: (1) en mængde multiple-choice-opgaver som tæller 75%, og (2) en traditionel skriftlig opgave som tæller 25%. Multiple-choice opgaverne besvares på opgaveformuleringen **som afleveres**. Den traditionelle opgave besvares på separat papir som **afleveres sammen med opgaveformuleringen**.

For hver opgave er angivet opgavens andel af det samlede eksamenssæt.

For multiple-choice delen, har hvert delspørgsmål præcist et rigtigt svar. For hvert delspørgsmål, kan du vælge **max ét svar** ved at afkrydse den tilsvarende rubrik. Et delspørgsmål bedømmes som følgende:

- Hvis du sætter kryds ved det rigtige svar, får du 1 point.
- Hvis du ikke sætter nogen krydser, får du 0 point.
- Hvis du sætter kryds ved et forkert svar, får du  $-\frac{1}{k-1}$  point, hvor  $k$  er antal svarmuligheder.

For en opgave med vægt  $v\%$  og med  $n$  delspørgsmål, hvor du opnår samlet  $s$  point, beregnes din besvarelse af opgaven som:

$$\frac{s}{n} \cdot v\%$$

Bemærk at det er muligt at få negative point for en opgave.

For den traditionelle opgave bedes du forsøge at svare så kort og præcist som muligt, og undgå at skrive ting der ikke er relevante for opgaven. Du må svare på dansk eller engelsk. Husk at skrive dit navne herunder:

Navn og studienummer: \_\_\_\_\_

## Multiple-choice del (75%)

**Opgave 1** (6%) I denne opgave angiver  $\log n$  2-tals-logaritmen af  $n$ ,  $\cdot$  er multiplikation og  $/$  er division. Angiv om hvert af følgende udsagn er sande eller falske.

	Ja	Nej
$17n$ er $O(4n)$ ?	<input type="radio"/>	<input type="radio"/>
$4n$ er $O(17n)$ ?	<input type="radio"/>	<input type="radio"/>
$n^3$ er $O(4n^3)$ ?	<input type="radio"/>	<input type="radio"/>
$n^{1/3}$ er $O(n^{1/4})$ ?	<input type="radio"/>	<input type="radio"/>
$n + n \log n$ er $O(n \log n)$ ?	<input type="radio"/>	<input type="radio"/>
$n^3/n^2$ er $O(n)$ ?	<input type="radio"/>	<input type="radio"/>
$n^{1/2} \cdot n^{2/3}$ er $O(n)$ ?	<input type="radio"/>	<input type="radio"/>
$1/(n \log n)$ er $O(1/n^2)$ ?	<input type="radio"/>	<input type="radio"/>
$\log(n^2)$ er $O(\log(3n))$ ?	<input type="radio"/>	<input type="radio"/>
$\log(n^3)$ er $O(5 \log n)$ ?	<input type="radio"/>	<input type="radio"/>
$(\log n)^2$ er $O(\log n)$ ?	<input type="radio"/>	<input type="radio"/>
$n^{1/10}$ er $O((\log n)^2)$ ?	<input type="radio"/>	<input type="radio"/>
$2^{10}$ er $O(\log \log n)$ ?	<input type="radio"/>	<input type="radio"/>
$3^n$ er $O(\log n)$ ?	<input type="radio"/>	<input type="radio"/>
$2^n$ er $O(4^n)$ ?	<input type="radio"/>	<input type="radio"/>
$4^n$ er $O(3^{2n})$ ?	<input type="radio"/>	<input type="radio"/>
$2^{3 \log n}$ er $O(n)$ ?	<input type="radio"/>	<input type="radio"/>
$2^{2 \log n \log n}$ er $O(\log n)$ ?	<input type="radio"/>	<input type="radio"/>
$n$ er $O(3^{\log n})$ ?	<input type="radio"/>	<input type="radio"/>
$n^3$ er $O(6^{\log n})$ ?	<input type="radio"/>	<input type="radio"/>

**Opgave 2** (6%) Betragt følgende max-heap, repræsenteret i et array  $A$ :

1	2	3	4	5	6	7	8	9	10
22	17	19	10	8	18	2	9	7	6

Angiv hvordan ovenstående max-heap ser ud efter indsættelse af elementet 20 (MAX-HEAP-INSERT( $A$ , 20))

- |    |    |    |    |    |    |   |   |   |    |    |
|----|----|----|----|----|----|---|---|---|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7 | 8 | 9 | 10 | 11 |
| 22 | 20 | 19 | 17 | 10 | 18 | 2 | 9 | 7 | 6  | 8  |
- |    |    |    |    |   |    |    |   |   |    |    |
|----|----|----|----|---|----|----|---|---|----|----|
| 1  | 2  | 3  | 4  | 5 | 6  | 7  | 8 | 9 | 10 | 11 |
| 22 | 17 | 20 | 10 | 8 | 18 | 20 | 9 | 7 | 6  | 2  |
- |    |    |    |    |   |    |   |   |   |    |    |
|----|----|----|----|---|----|---|---|---|----|----|
| 1  | 2  | 3  | 4  | 5 | 6  | 7 | 8 | 9 | 10 | 11 |
| 22 | 17 | 19 | 10 | 8 | 18 | 2 | 9 | 7 | 6  | 22 |
- |    |    |    |    |    |    |   |   |   |    |    |
|----|----|----|----|----|----|---|---|---|----|----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7 | 8 | 9 | 10 | 11 |
| 22 | 20 | 19 | 10 | 17 | 18 | 2 | 9 | 7 | 6  | 8  |

**Opgave 3** (6%) Antag vi bruger MERGE-SORT til at sortere følgende array:

1	2	3	4	5	6	7	8
3	7	1	2	8	9	4	6

For hver af følgende par af elementer, angiv om MERGE-SORT på ovenstående array vil sammenligne de to elementer som en del af sin udførelse.

Ja    Nej

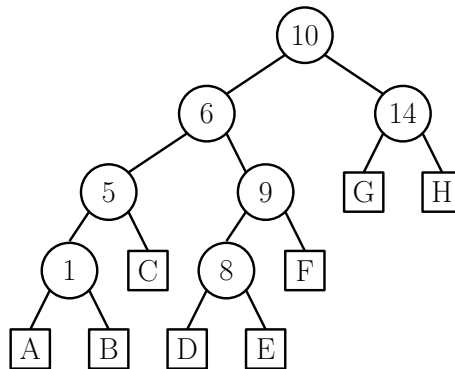
**Opgave 4** (5%) Betragt RADIX-SORT anvendt på nedenstående liste af tal ( $d = 4, k = 4$ ).

0331      1122      2210      3100      0031      1110

Angiv den delvist sortede liste efter at radix-sort har sorteret tallene efter de **to** mindst betydende cifre.

- 3100      2210      1110      1122      0331      0031
- 3100      2210      1110      0331      0031      1122
- 3100      1110      2210      1122      0031      0331
- 0031      0331      1110      1122      2210      3100

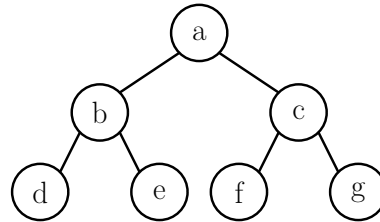
**Opgave 5** (6%) Betragt følgende ubalancerede søgetræ:



Angiv i hvilke blade A-H elementerne 4, 12, 7, 0, 3 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående syv elementer).

	A	B	C	D	E	F	G	H
Insert(4)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Insert(12)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Insert(7)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Insert(0)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Insert(3)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

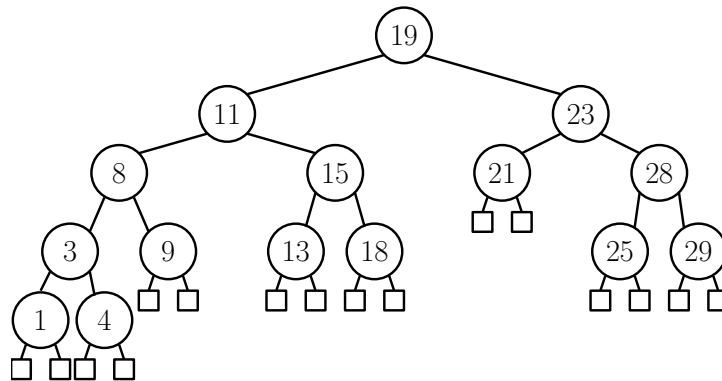
**Opgave 6** (6%) Betragt følgende binære søgetræ indeholdende knuderne  $a, b, c, d, e, f, g$ :



Nøglerne hørende til knuderne vises ikke på figuren overfor, men det antages at det er et gyldigt søgetræ. Antag at vi nu laver en venstre-rotation (LEFT-ROTATE) på kanten imellem  $a$  og  $c$ . Angiv for hvert af følgende udsagn om det er sandt for søgetræet efter rotationen.

	Ja	Nej
$e$ er højre barn til $b$ ( $b.right = e$ )	<input type="radio"/>	<input type="radio"/>
$f$ er venstre barn til $g$ ( $g.left = f$ )	<input type="radio"/>	<input type="radio"/>
$b$ er venstre barn til $c$ ( $c.left = b$ )	<input type="radio"/>	<input type="radio"/>
$a$ er højre barn til $b$ ( $b.right = a$ )	<input type="radio"/>	<input type="radio"/>

**Opgave 7** (6%) For hver af nedenstående delmængder, angiv om følgende binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde



	Ja	Nej
3, 8, 15, 23, 25, 29	<input type="radio"/>	<input type="radio"/>
3, 9, 11, 13, 18, 23, 25, 29	<input type="radio"/>	<input type="radio"/>
1, 4, 8, 15, 25, 29	<input type="radio"/>	<input type="radio"/>
3, 11, 28	<input type="radio"/>	<input type="radio"/>

**Opgave 8** (5%) I følgende hashtabel er anvendt *quadratic probing* med hashfunktionen  $h(k, i) = (k^2 + i + 2i^2) \pmod{11}$ .

0	1	2	3	4	5	6	7	8	9	10
	10		6	12	7				8	

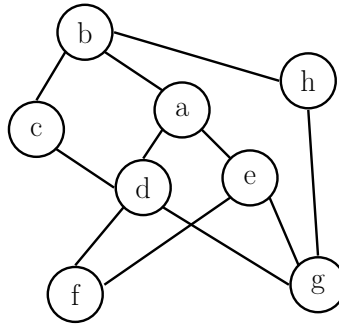
Angiv hvordan hashtabellen ser ud efter elementerne 4, 2 og 3 er blevet tilføjet (indsat i den rækkefølge, altså først 4, så 2 og tilsidst 3).

- | 0 | 1  | 2 | 3 | 4  | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|---|---|----|---|---|---|---|---|----|
|   | 10 |   | 6 | 12 | 7 |   | 2 | 4 | 8 | 3  |
- | 0 | 1  | 2 | 3 | 4  | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|---|---|----|---|---|---|---|---|----|
| 3 | 10 |   | 6 | 12 | 7 |   | 4 |   | 8 | 2  |
- | 0 | 1  | 2 | 3 | 4  | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|---|---|----|---|---|---|---|---|----|
|   | 10 |   | 6 | 12 | 7 |   | 4 | 3 | 8 | 2  |
- | 0 | 1  | 2 | 3 | 4  | 5 | 6 | 7 | 8 | 9 | 10 |
|---|----|---|---|----|---|---|---|---|---|----|
| 3 | 10 |   | 6 | 12 | 7 |   | 2 | 4 | 8 |    |

**Opgave 9** (6%) Hver af følgende rekursionsligninger (recurrences) har basistilfældet  $T(1) = 1$ . Angiv for hver ligning, hvad løsningen er:

- $T(n) = 3T(n/3) + n^2$      $\Theta(n^2)$      $\Theta(n^2 \log n)$      $\Theta(n)$   
 $T(n) = 2T(n/3) + \sqrt{n}$      $\Theta(\sqrt{n})$      $\Theta(\sqrt{n} \log n)$      $\Theta(n^{\log_3 2})$      $\Theta(n^{\log_2 3})$   
 $T(n) = 5T(n/2) + n^2$      $\Theta(n^2)$      $\Theta(n^2 \log n)$      $\Theta(n^{\log_2 5})$      $\Theta(n^{\log_5 2})$   
 $T(n) = 9T(n/3) + n^2$      $\Theta(n^2)$      $\Theta(n^2 \log n)$      $\Theta(n^{\log_9 3})$

**Opgave 10** (5%) Betragt følgende uorienterede graf:

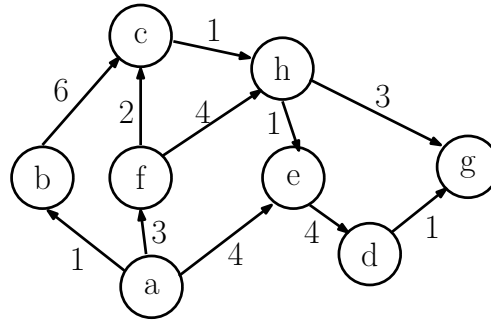


Angiv for hver af følgende delmængder af kanter, hvorvidt man ved at udføre en bredde-først søgning (breadth-first search BFS) fra knuden  $a$ , kan få netop denne mængde af kanter som det resulterende bredde-først træ.

	Ja	Nej
$\{a,b\}, \{a,d\}, \{a,e\}, \{b,c\}, \{b,h\}, \{d,f\}, \{d,g\}$	<input type="radio"/>	<input type="radio"/>
$\{a,b\}, \{a,d\}, \{a,e\}, \{b,h\}, \{d,c\}, \{e,f\}, \{e,g\}$	<input type="radio"/>	<input type="radio"/>
$\{a,b\}, \{a,d\}, \{a,e\}, \{b,c\}, \{b,h\}, \{d,f\}, \{e,g\}$	<input type="radio"/>	<input type="radio"/>
$\{a,b\}, \{a,d\}, \{a,e\}, \{d,c\}, \{e,f\}, \{e,g\}, \{g,h\}$	<input type="radio"/>	<input type="radio"/>



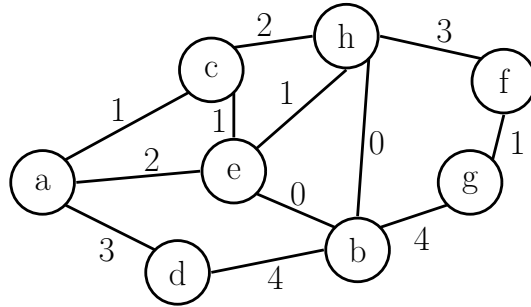
**Opgave 11** (6%) Betragt følgende vægtede orienterede graf:



Antag vi kører Dijkstra's korteste veje algoritme (single source shortest paths SSSP), med knuden  $a$  som source. Efter 4 iterationer vil knuderne  $a$ ,  $b$ ,  $f$  og  $e$  være hevet ud af prioritetskøen (i den rækkefølge), og der er kaldt RELAX på deres udgående kanter. Angiv for hvert af følgende afstandsestimater, hvorvidt det svarer til det estimat Dijkstra's algoritme har på det tidspunkt, altså lige inden den 5. knude hives ud af prioritetskøen.

- |           | Ja                    | Nej                   |
|-----------|-----------------------|-----------------------|
| $c.d = 2$ | <input type="radio"/> | <input type="radio"/> |
| $h.d = 7$ | <input type="radio"/> | <input type="radio"/> |
| $d.d = 8$ | <input type="radio"/> | <input type="radio"/> |
| $g.d = 9$ | <input type="radio"/> | <input type="radio"/> |

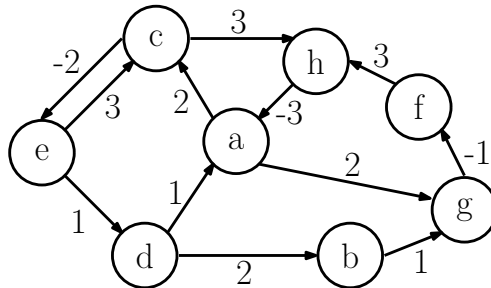
**Opgave 12** (6%) Betragt følgende vægtede uorienterede graf:



Angiv for hver af følgende kanter, hvorvidt netop denne kant er med i det (unikke) minimum udspændende træ (minimum spanning tree MST) for grafen.

	Ja	Nej
{h,f}	<input type="radio"/>	<input type="radio"/>
{e,b}	<input type="radio"/>	<input type="radio"/>
{e,h}	<input type="radio"/>	<input type="radio"/>
{b,h}	<input type="radio"/>	<input type="radio"/>
{a,e}	<input type="radio"/>	<input type="radio"/>

**Opgave 13** (6%) Betragt følgende vægtede orienterede graf:



Betragt den korteste vej i grafen ovenfor i blandt hver af følgende knudepar: (1) fra  $a$  til  $c$ , (2) fra  $h$  til  $d$ , (3) fra  $a$  til  $d$ , (4) fra  $b$  til  $a$ , (5) fra  $c$  til  $g$ . Angiv hvilken af de fem korteste veje, som er kortest:

- Fra  $a$  til  $c$
- Fra  $h$  til  $d$
- Fra  $a$  til  $d$
- Fra  $b$  til  $a$
- Fra  $c$  til  $g$

## Traditionel del (25%)

En træt forelæser har skrevet en lecture note til de studerende i FADS kurset, men den stakkels forelæser har været så træt at han har glemt at sætte mellemrum i hele teksten:

Idenneforelæsningskalvilæreomdynamiskprogrammering

Din opgave er at bruge en ordbog samt dynamisk programmering til at afgøre om det er muligt at indsætte mellemrum således at teksten består af en sekvens af ord fra ordbogen, a. la.:

I denne forelæsning skal vil ære om dynamisk programmering

Mere formelt er vi givet en streng  $T = \sigma_1 \cdots \sigma_n$  af længde  $n$ , hvor hvert  $\sigma_i$  er et symbol i blandt  $\{a, \dots, \grave{a}, A, \dots, \grave{A}\}$ . Vi er endvidere givet en *ordbog*  $\mathcal{D} = \{S_1, \dots, S_k\}$ , bestående af strenge  $S_i$  over alfabetet  $\{a, \dots, \grave{a}, A, \dots, \grave{A}\}$ . Vi lader  $m$  være den *samlede* længde af alle strenge i  $\mathcal{D}$ , altså  $m = \sum_{i=1}^k |S_i|$  hvor  $|S_i|$  giver længden af strengen  $S_i$ . Vi lader  $k$  være antallet af strenge i  $\mathcal{D}$ . Din opgave er at afgøre om det er muligt at skrive strengen  $T$  som en konkatenering af strenge fra  $\mathcal{D}$  (den samme streng fra  $\mathcal{D}$  må gerne bruges flere gange). Et eksempel kunne være:

$T =$  Idenneforelæsningskalvilæreomdynamiskprogrammering

og

$\mathcal{D} = \{\text{alle, mus, denne, ære, dynamisk, om, forelæsning, I, programmering, skal, vil, kvinde}\}$

I dette eksempel er svaret ja, da vi kan skrive strengen  $T$  som konkateneringen af

I denne forelæsning skal vil ære om dynamisk programmering

hvilket udelukkende er strenge fra  $\mathcal{D}$ . Hvis vi i stedet havde strengen

$T' =$  skalmuslæredynamiskprogrammering

med den samme ordbog  $\mathcal{D}$ , så ville svaret være nej da det ikke er muligt at matche andet end

skal mus

fra starten af  $T'$  med ord fra  $\mathcal{D}$ .

Hvis vi lader  $OPT(i)$  udtrykke hvorvidt det er muligt at sætte mellemrum i prefixet  $\sigma_1 \cdots \sigma_i$  af strengen  $T = \sigma_1 \cdots \sigma_n$ , således at alle de resulterende ord kommer fra ordbogen  $\mathcal{D} = \{S_1, \dots, S_k\}$ , så kan  $OPT(i)$  beskrives således:

$$OPT(i) = \bigvee_{j=1}^k \left\{ |S_j| \leq i \wedge OPT(i - |S_j|) \wedge \left( \bigwedge_{\ell=1}^{|S_j|} \{S_j(\ell) = \sigma_{i-|S_j|+\ell}\} \right) \right\}$$

I udtrykket for  $OPT(i)$  er  $\{S_j(\ell) = \sigma_{i-|S_j|+\ell}\}$  sandt såfremt det  $\ell$ 'te tegn i  $S_j$  er lig det  $(i - |S_j| + \ell)$ 'te tegn i  $T$ . Notationen  $\vee$  betyder *eller* og notationen  $\wedge$  betyder *og*.  $\bigvee_{j=1}^k$  er altså sandt såfremt mindst én værdi af  $j$  får udtrykket inde i de yderste  $\{\}$  til at blive sandt. Det indre udtryk  $\bigwedge_{\ell=1}^{|S_j|}$  er sandt kun hvis alle værdier af  $\ell$  får  $\{S_j(\ell) = \sigma_{i-|S_j|+\ell}\}$  til at blive sandt.

For en streng  $T$  af længde  $n$  og en ordbog  $\mathcal{D}$  angiver værdien  $OPT(n)$  altså hvorvidt det er muligt at sætte mellemrum i  $T$  således at alle de resulterende ord er i ordbogen.

**Opgave 14** (8%) Udfyld nedenstående tabel for  $OPT(i)$  når strengen er

$$T = \text{nuerdusnartfærdig}$$

og ordbogen er

$$\mathcal{D} = \{\text{dus, du, nu, er, snart, ær, færdig}\}$$

Brug  $T$  for True/Sand/Ja og  $F$  for False/Falsk/Nej:

$T$	n	u	e	r	d	u	s	n	a	r	t	f	æ	r	d	i	g
$i$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$OPT(i)$	F	T															

**Opgave 15** (10%) Angiv en algoritme baseret på dynamisk programmering, der givet en tekst  $T = \sigma_1 \cdots \sigma_n$  af længde  $n$  og en ordbog  $\mathcal{D} = \{S_1, \dots, S_k\}$  med  $\sum_{i=1}^k |S_i| = m$ , beregner hvorvidt det er muligt at sætte mellemrum i  $T$  således at de resulterende ord alle er i  $\mathcal{D}$ , dvs. algoritmen beregner  $OPT(n)$ . Angiv algoritmens udførelsestid. Undlad at skrive udtryk som  $\bigvee_{j=1}^k$  i din pseudokode - brug i stedet for-løkker eller lignende.

**Opgave 16** (7%) Udvid algoritmen til at udskrive strengen  $T$  med mellemrum således at alle de resulterende ord optræder i  $\mathcal{D}$ . Hvis dette ikke er muligt skal algoritmen udskrive "Not possible". Angiv algoritmens udførelsestid.