

Skriftlig Eksamen
Algoritmer og Datastrukturer 1

Datalogisk Institut
Aarhus Universitet

Mandag den 22. marts 2004, kl. 9.00–11.00

Navn Gerth Stølting Brodal

Årskort 19890616

Dette eksamenssæt består af en kombination af små skriftlige opgaver og multiple-choice-opgaver. Opgaverne besvares på opgaveformuleringen **som afleveres**.

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

For multiple-choice-opgaver gælder følgende. Hvert delspørgsmål har præcist et svar. For hvert delspørgsmål, kan du vælge et eller flere svar ved at afkrydse de tilsvarende rubrikker. Et multiple-choice-delspørgsmål bedømmes som følgende:

- Hvis du kun sætter kryds ved det rigtige svar får du 1 point.
- Hvis du sætter a krydser, hvor det ene er det rigtige, får du $1/a$ point.
- Hvis du vælger a krydser, hvor ingen er det rigtige, får du 0 point.
- Hvis du ikke sætter nogen krydser, får du $1/k$ point, hvor k er antal svarmuligheder.

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

$$\max \left\{ 0, \frac{s - n/k}{n - n/k} \right\} \cdot v \%$$

Opgave 1 (6 %)

	Ja	Nej
$3n$ er $O(n)$?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
\sqrt{n} er $O(n^2)$?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$n \cdot \log n$ er $O(n)$?	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$n + \sqrt{n}$ er $O(n \cdot \log n)$?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$5n + 3n^2$ er $O(8n)$?	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Opgave 2 (6 %)

Opskriv følgende funktioner efter stigende orden med hensyn til O -notationen:

$7n^2 + 3n$
 $n + \sqrt{n}$
 $n \cdot \log n$
 2^n

Svar: $n + \sqrt{n}$ $n \cdot \log n$ $7n^2 + 3n$ 2^n

Opgave 3 (6 %)

Angiv for hver af nedenstående algoritmer udførelstiden som funktion af n i O -notation.

Algoritme Loop1(n)

```
 $x \leftarrow 0$   
for  $i \leftarrow 1$  to  $n$  do  
    for  $j \leftarrow 1$  to  $n$  do  
         $x \leftarrow x + i * j$ 
```

Algoritme Loop2(n)

```
 $x \leftarrow 0$   
for  $i \leftarrow 1$  to  $n$  do  
    for  $j \leftarrow 1$  to  $i$  do  
         $x \leftarrow x + i * j$ 
```

Algoritme Loop3(n)

```
 $x \leftarrow 0$   
 $i \leftarrow 1$   
while ( $i \leq n$ )  
     $i \leftarrow i * 2$   
     $x \leftarrow x + 1$ 
```

Svar Loop1: $O(n^2)$

Svar Loop2: $O(n^2)$

Svar Loop3: $O(\log n)$

Opgave 4 (5%)

Tillader nedenstående datastrukturer indsættelser af nye elementer i tid $O(\log n)$?

	Ja	Nej
Usorteret kædet liste	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Heap	<input checked="" type="checkbox"/>	<input type="checkbox"/>
AVL-træ	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Tillader nedenstående datastrukturer at slette elementet med den mindste nøgle i tid $O(\log n)$?

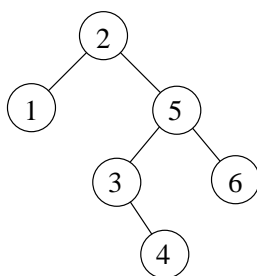
	Ja	Nej
Usorteret kædet liste	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Heap	<input checked="" type="checkbox"/>	<input type="checkbox"/>
AVL-træ	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Tillader nedenstående datastrukturer at finde et element med en given nøgle i tid $O(\log n)$?

	Ja	Nej
Usorteret kædet liste	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Heap	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AVL-træ	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Opgave 5 (6%)

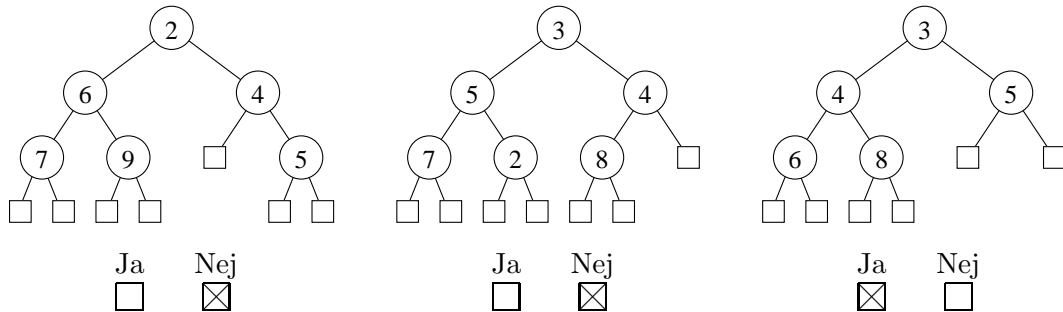
Angiv for hver af nedenstående sekvenser om den angiver et preorder, postorder, eller et inorder gennemløb af nedenstående træ.



	Preorder	Postorder	Inorder	Hverken preorder, postorder, eller inorder
1 4 3 6 5 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1 4 3 5 6 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
1 2 3 4 5 6	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2 1 5 3 4 6	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6 5 4 3 2 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

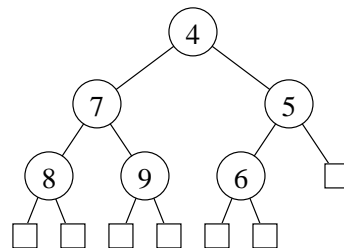
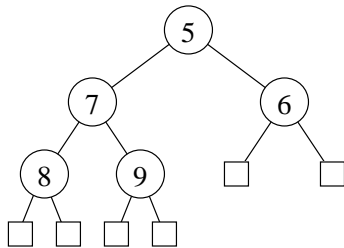
Opgave 6 (5%)

Angiv for hver af nedenstående binære træer om det er en lovlig heap.



Opgave 7 (5%)

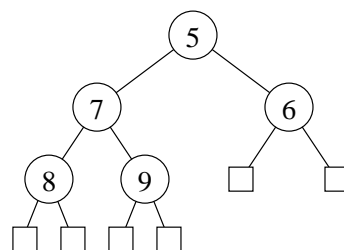
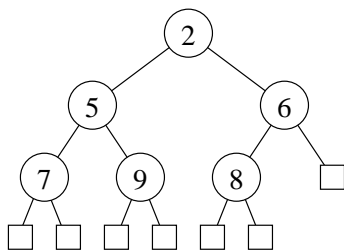
Tegn hvordan nedenstående heap ser ud efter indsættelse af elementet 4.



Svar: _____

Opgave 8 (6%)

Tegn hvordan nedenstående heap ser ud efter en removeMin operation.



Svar: _____

Opgave 9 (6%)

Nedenstående er en hashtabel hvor der er anvendt *linear probing*. Den anvendte hashfunktion er $h(k) = k * 3 \text{ mod } 17$. Tegn hvordan hashtabellen ser ud efter at $k = 25$ indsættes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17						2	8	19	14		15	4	10		5	

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17						2	8	19	14	25	15	4	10		5	

Svar: _____

Opgave 10 (6%)

Nedenstående er en hashtabel hvor der er anvendt *linear probing*. Den anvendte hashfunktion er $h(k) = k * 3 \text{ mod } 17$. Tegn hvordan hashtabellen ser ud efter at $k = 2$ slettes.

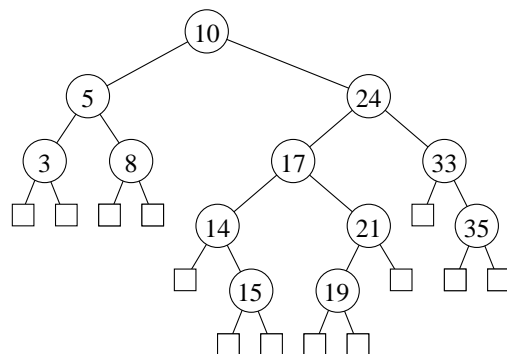
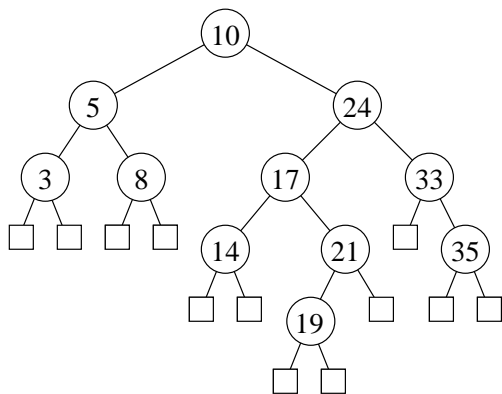
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17						2	8	19	14		15	4	10		5	

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17						19	8	14			15	4	10		5	

Svar: _____

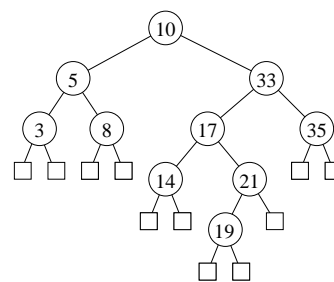
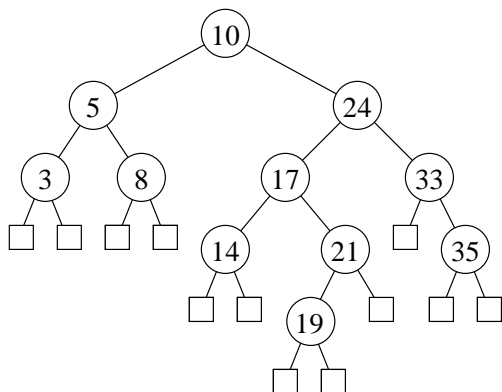
Opgave 11 (6%)

Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter indsættelse af elementet 15.

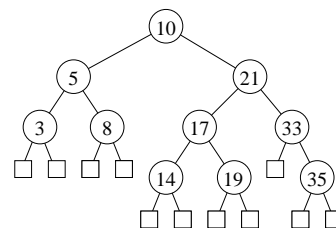


Svar: _____

Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter slettelse af elementet 24.



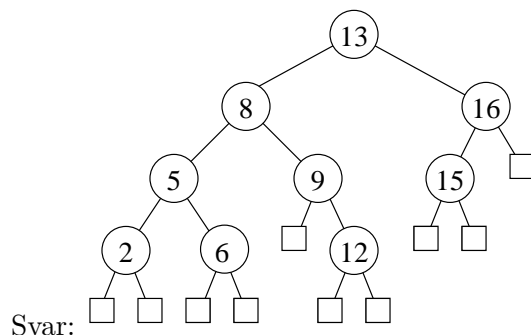
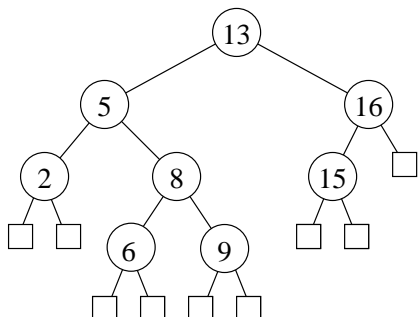
eller



Svar: _____

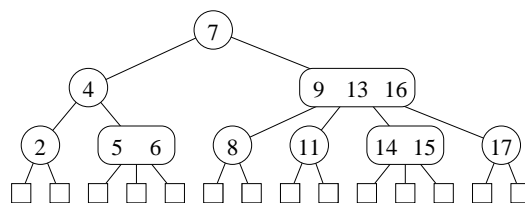
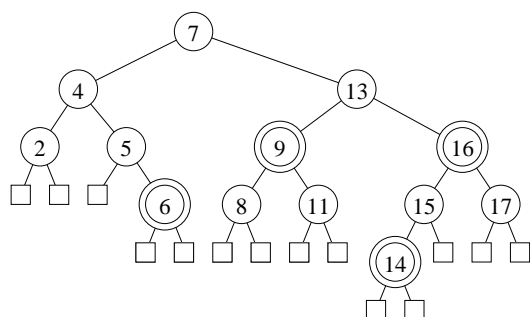
Opgave 12 (6%)

Tegn hvordan nedenstående AVL-træ ser ud efter indsættelse af elementet 12.



Opgave 13 (5%)

Nedenstående er et rød-sort træ (dobbeltcirkler angiver røde knuder). Tegn det korresponderende (2,4) træ.



Opgave 14 (6%)

Betragt den træ-baserede union-find datastruktur, hvor man anvender union-by-size heuristikken. Hvad er højden af træet der repræsenterer en mængde med n elementer, udtrykt i O -notation?

Svar: $O(\log n)$

Opgave 15 (6%)

Angiv 3 sorteringsalgoritmer til sortering af n elementer, som har forventet eller worst-case $O(n \cdot \log n)$ udførselstid.

Svar 1: QuickSort

Svar 2: MergeSort

Svar 3: HeapSort

Opgave 16 (6 %)

Algoritme Loop(n)
Inputbetingelse : $n \geq 1$
Outputkrav : -
Metode : $i \leftarrow 1$;
 while $i < n$ **do**
 if ($2 * i \leq n$)
 $i \leftarrow 2 * i$
 else
 $i \leftarrow i + 1$

For hver af nedenstående funktioner, angiv om de er en termineringsfunktion for ovenstående algoritme.

	Ja	Nej
$\mu(i, n) = n$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, n) = i$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, n) = n - i$	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$\mu(i, n) = \log n - \log i$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, n) = 2^n - i$	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Opgave 17 (8 %)

Nedenstående algoritme beregner summen af elementerne i et array. For at vise gyldigheden af algoritmen skal I_i og I_s være invarianter omkring s og i . Angiv invarianter hvormed gyldigheden af algoritmen kan bevises (bevis for invarianterne kræves ikke).

Algoritme Sum(A)
Inputbetingelse : Array A af n heltal, $n \geq 1$
Outputkrav : $s = \sum_{j=0}^{n-1} A[j]$
Metode : $s \leftarrow 0$;
 $i \leftarrow 0$;
 $\{I_i \wedge I_s\}$ **while** $i < n$ **do**
 $s \leftarrow s + A[i]$;
 $i \leftarrow i + 1$

Svar I_i : $0 \leq i \leq n$

Svar I_s : $s = \sum_{j=0}^{i-1} A[j]$

For at kunne bevise at algoritmen terminerer, kræves en passende termineringsfunktion. Angiv en termineringsfunktion (bevis for at termineringsfunktionen har de nødvendige egenskaber kræves ikke).

Svar μ : $n - i$