

**Opgave 1 (4 %)**

	Ja	Nej
$7n^2$ er $O(10n \cdot \log n)$ ?	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$3^n$ er $O(n^4)$ ?	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$n^2 + 7n$ er $O(n^3)$ ?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$n \cdot \log n$ er $O(n^2)$ ?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$\sqrt{n} \cdot \log n$ er $O(n)$ ?	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Opgave 2 (4 %)**

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

$$4^n \cdot n$$
$$3n$$
$$n^2 / \log n$$
$$5^n$$
$$\sqrt{n} \cdot (\log n)^2$$

Svar: \_\_\_\_\_  $\sqrt{n} \cdot (\log n)^2$     $3n$     $n^2 / \log n$     $4^n \cdot n$     $5^n$

**Opgave 3 (4 %)**

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  $i$  do  
     $x \leftarrow x + 1$ 
```

**Algoritme Loop2( $n$ )**

```
 $i \leftarrow 1$   
while  $i \leq n$  do  
   $i \leftarrow 2 * i$ 
```

**Algoritme Loop3( $n$ )**

```
 $i \leftarrow 1$   
 $x \leftarrow 0$   
while  $i \leq n$  do  
  for  $j \leftarrow 1$  to  $i$  do  
     $x \leftarrow x + 1$   
   $i \leftarrow 2 * i$ 
```

Svar Loop1: \_\_\_\_\_  $O(n^2)$

Svar Loop2: \_\_\_\_\_  $O(\log n)$

Svar Loop3: \_\_\_\_\_  $O(n)$

**Opgave 4 (4%)**

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

```
Algoritme Rec1( $n$ )  
if  $n \leq 1$  then  
    return 1  
else  
    return  $1 + \text{Rec1}(n - 1)$ 
```

```
Algoritme Rec2( $n$ )  
if  $n \leq 1$  then  
    return 1  
else  
    return  $\text{Rec2}(n/2) + \text{Rec2}(n/2)$ 
```

```
Algoritme Rec3( $n$ )  
if  $n \leq 1$  then  
    return 1  
else  
    return  $1 + \text{Rec3}(n/2)$ 
```

Svar Rec1:                      $O(n)$                     

Svar Rec2:                      $O(n)$                     

Svar Rec3:                      $O(\log n)$                     

**Opgave 5 (4%)**

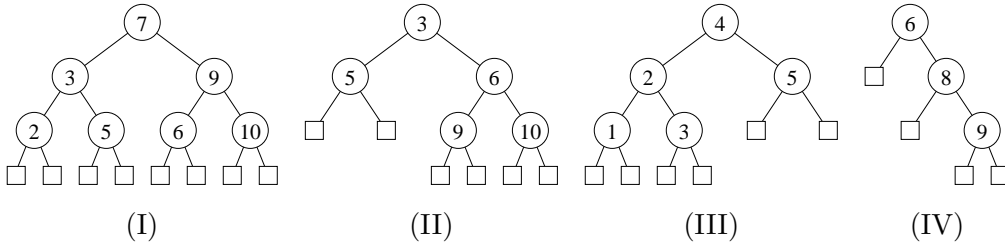
Betragt radix-sort anvendt på nedenstående liste af tal ( $d = 4$ ,  $N = 10$ ). Angiv den delvist sorterede liste efter at radix-sort har sorteret tallene efter de to mindst betydende cifre.

1981 1324 1715 1011 2414 1314

Svar:                     1011 2414 1314 1715 1324 1981

**Opgave 6 (4%)**

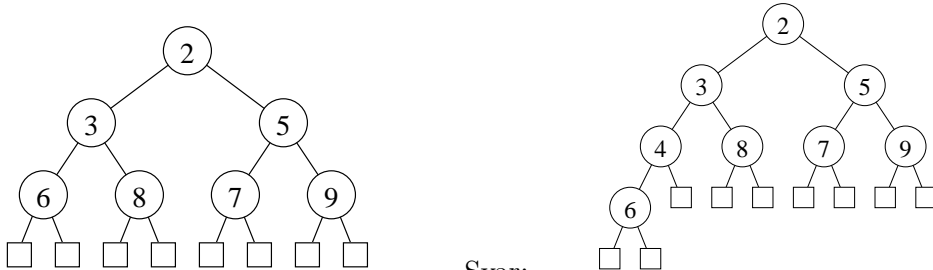
Angiv for hver af nedenstående binære træer om det er et lovligt ubalanceret binært søgetræ eller opfylder heap-order med hensyn til minimum.



	Søgetræ	Heap-order	Både søgetræ og heap-order	Hverken søgetræ eller heap-order
(I)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
(II)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
(III)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
(IV)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Opgave 7 (4%)**

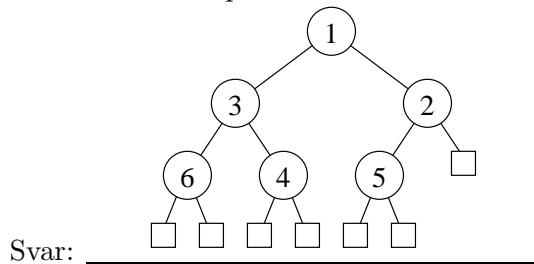
Tegn hvordan nedenstående binære heap ser ud efter indsættelse af elementet 4.



Svar: \_\_\_\_\_

**Opgave 8 (4%)**

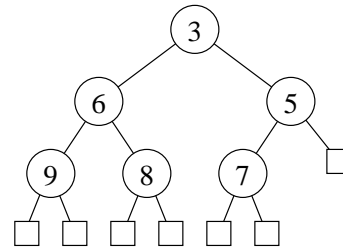
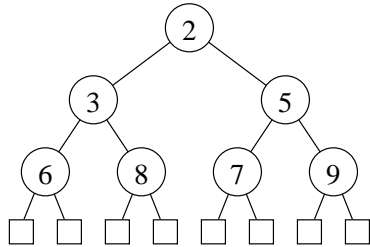
Tegn den binære heap efter indsættelse af elementerne 6, 5, 4, 3, 2, 1 i en heap i den givne rækkefølge, startende med den tomme heap.



Svar: \_\_\_\_\_

**Opgave 9 (4%)**

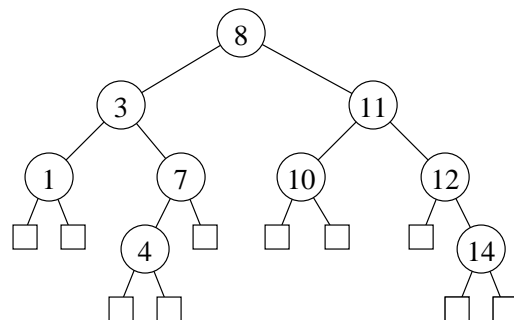
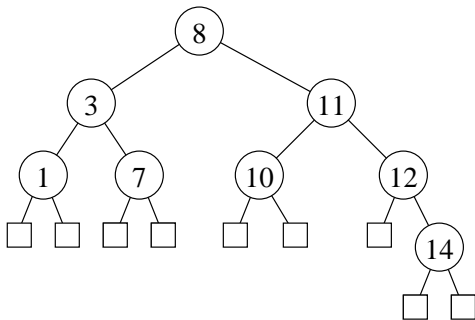
Tegn hvordan nedenstående binære heap ser ud efter en removeMin operation.



Svar: \_\_\_\_\_

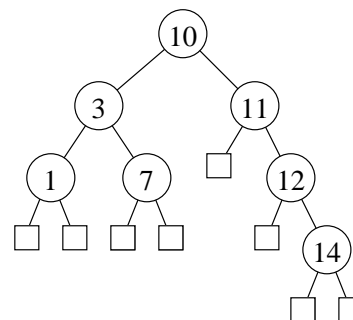
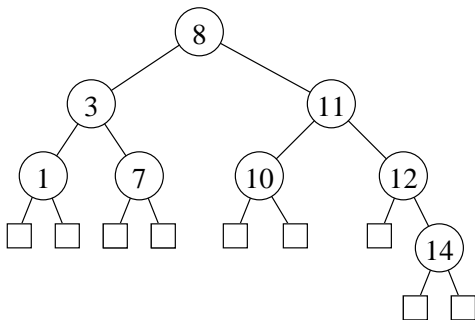
**Opgave 10 (4%)**

Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter indsættelse af elementet 4.



Svar: \_\_\_\_\_

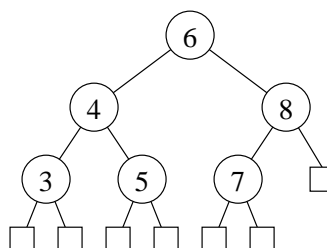
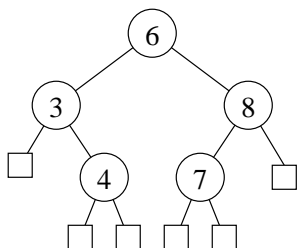
Tegn hvordan nedenstående ubalancerede binære søgetræ ser ud efter slettelse af elementet 8.



Svar: \_\_\_\_\_

**Opgave 11 (4%)**

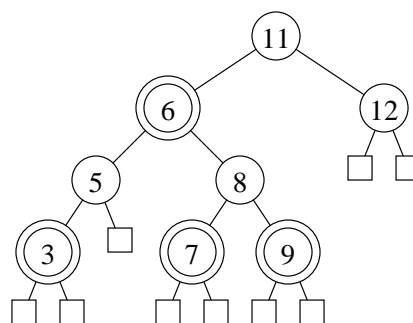
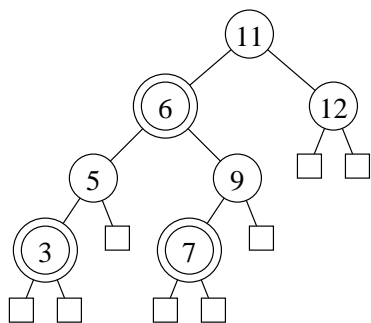
Tegn hvordan nedenstående AVL-træ ser ud efter indsættelse af elementet 5.



Svar: \_\_\_\_\_

**Opgave 12 (4%)**

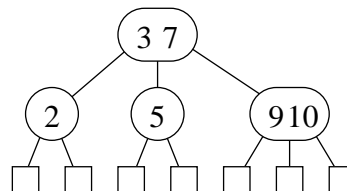
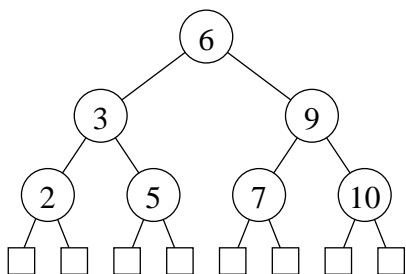
Tegn hvordan nedenstående rød-sort træ (dobbeltcirkler angiver røde knuder) ser ud efter indsættelse af elementet 8.



Svar: \_\_\_\_\_

**Opgave 13 (4%)**

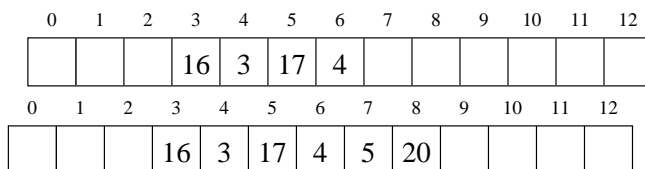
Tegn hvordan nedenstående (2,4)-træ ser ud efter slettelse af elementet 6.



Svar: \_\_\_\_\_

**Opgave 14 (4%)**

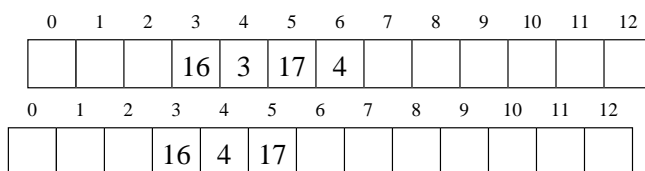
Nedenstående er en hashtabel hvor der er anvendt *linear probing*. Den anvendte hashfunktion er  $h(k) = k \bmod 13$ . Tegn hvordan hashtabellen ser ud efter at  $k = 5$  og  $k = 20$  indsættes i den givne rækkefølge.



Svar: \_\_\_\_\_

**Opgave 15 (4%)**

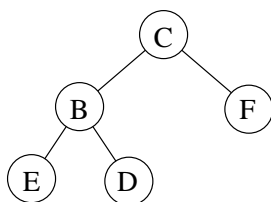
Nedenstående er en hashtabel hvor der er anvendt *linear probing*. Den anvendte hashfunktion er  $h(k) = k \bmod 13$ . Tegn hvordan hashtabellen ser ud efter at  $k = 3$  slettes fra hashtabellen (der må ikke bruges dummy elementer).



Svar: \_\_\_\_\_

**Opgave 16 (4%)**

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

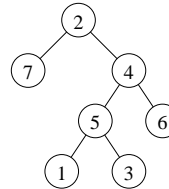


	Preorder	Postorder	Inorder	Hverken preorder, postorder, eller inorder
C B E D F	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C B F E D	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
E D B F C	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
E B D C F	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
E D B C F	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Opgave 17 (4%)**

Tegn det binære træ hvor alle knuder har 0 eller 2 børn og hvor henholdsvis et preorder og postorder gennemløb giver:

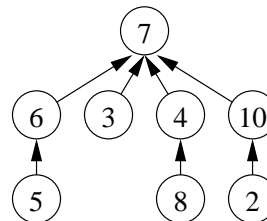
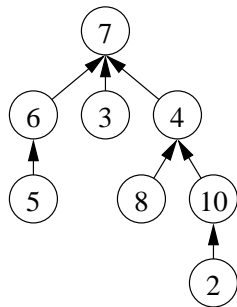
Preorder : 2 7 4 5 1 3 6  
 Postorder : 7 1 3 5 6 4 2



Svar: \_\_\_\_\_

**Opgave 18 (4%)**

Tegn hvordan nedenstående union-find datastruktur ser ud efter FIND(10), når der anvendes stikomprimering.



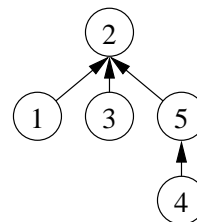
Svar: \_\_\_\_\_

**Opgave 19 (4%)**

Tegn hvordan nedenstående union-find datastruktur ser ud efter

UNION(UNION(UNION(1,2),3),UNION(4,5))

når der anvendes union-by-size heuristikken.



Svar: \_\_\_\_\_

**Opgave 20 (4%)**

Angiv for randomiseret quick-select den forventede og worst-case udførselstid.

Forventet : \_\_\_\_\_  $O(n)$  \_\_\_\_\_

Worst-case : \_\_\_\_\_  $O(n^2)$  \_\_\_\_\_

**Transitionssystem** Countdown( $i, j$ )  
Configurations:  $\{[i, j] \mid i, j \geq 0\}$   
 $[i, j] \triangleright [i - 1, j + 2] \quad \mathbf{if} \quad i > 0$   
 $[i, j] \triangleright [i, j - 1] \quad \mathbf{if} \quad j > 0$

**Opgave 21 (4%)**

For hver af nedenstående udsagn, angiv om de er en invariant for ovenstående transitionssystem Countdown. Startkonfigurationen antages at være  $[i_0, j_0]$ .

	Ja	Nej
$0 \leq i \leq i_0$	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$0 \leq j \leq j_0$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$2i + j \leq 2i_0 + j_0$	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$j \leq 2i$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$j - j_0 \leq 2i_0$	<input checked="" type="checkbox"/>	<input type="checkbox"/>

**Opgave 22 (4%)**

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

	Ja	Nej
$\mu(i, j) = i + j$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, j) = i$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, j) = i \cdot j$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, j) = \max\{i, j\}$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$\mu(i, j) = 3i + j$	<input checked="" type="checkbox"/>	<input type="checkbox"/>



**Algoritme** Loop( $n$ )  
Inputbetingelse : heltal  $n \geq 1$   
Outputkrav : –  
Metode :  $i \leftarrow 1$ ;  
 $j \leftarrow 0$ ;  
 $\{I\}$  **while**  $j < n$  **do**  
 $j \leftarrow j + i$ ;  
 $i \leftarrow i + i$

**Opgave 23 (4%)**

For hver af nedenstående udsagn, angiv om de er en invariant  $I$  for ovenstående algoritme Loop.

	Ja	Nej
$0 \leq j \leq n$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$j < i$	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$j = i - 1$	<input checked="" type="checkbox"/>	<input type="checkbox"/>
$j = 2i - 2$	<input type="checkbox"/>	<input checked="" type="checkbox"/>
$i = 2^j$	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Opgave 24 (4%)**

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

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

**Opgave 25 (4%)**

Nedenstående algoritme beregner  $n^m$ . For at vise gyldigheden af algoritmen skal  $I_n$ ,  $I_m$  og  $I_r$  være invarianter omkring variablerne  $n$ ,  $m$  og  $r$ . Angiv invarianter hvormed gyldigheden af algoritmen kan bevises (bevis for invarianterne kræves ikke).

**Algoritme** Exp( $n, m$ )  
Inputbetingelse : heltal  $n, m \geq 1$   
Outputkrav :  $r = n^m$   
Metode :  $r = 1$ ;  
           $\{I_n \wedge I_m \wedge I_r\}$  **while**  $m \geq 1$  **do**  
                                   $m \leftarrow m - 1$ ;  
                                   $r \leftarrow r * n$

Svar  $I_n$ :                                    $n = n_0$                                   

Svar  $I_m$ :                                    $0 \leq m \leq m_0$                                   

Svar  $I_r$ :                                    $r = n_0^{m_0 - m}$                                   

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  $\mu$ :                                    $m$