

EKSAMEN

Algoritmer og Datastrukturer

(indsæt dato og tid)

Institut for Datalogi, Naturvidenskabelige Fakultet, Aarhus Universitet

Antal sider i opgavesættet (incl. forsiden): 126

Tilladte medbragte hjælpemidler: **Ingen**

Studienummer :

Navn :

compute score (max 920 point)

Vejledning og pointgivning

Dette eksamenssæt består af en mængde multiple-choice-opgaver.

Opgaverne besvares på opgaveformuleringen **som afleveres**.

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

Hvert delspørgsmål har præcis et rigtigt svar.

For hvert delspørgsmål må 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.

Opgave 1 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n .

Ja Nej

$$n + 4^{\log n} \text{ er } O(1)$$

$$n^{0.01} \text{ er } O(n\sqrt{n})$$

$$\log n + n^{2/3} \cdot n^{1/3} \text{ er } O((\log n)^3)$$

$$(\log n)^3 \text{ er } O(3^n)$$

$$n \text{ er } O(\log n^2)$$

$$n \cdot \log n \text{ er } O(n^3)$$

$$2\sqrt{n} \cdot \log n + n^2 \text{ er } O(\sqrt{n} \cdot \log n)$$

$$\log n \text{ er } O(3^n)$$

$$n^3 \text{ er } O(n^{2/3})$$

$$7(\log n)^2 + \sqrt{n} \cdot \log n \text{ er } \Omega(n \cdot \log n)$$

$$n! \text{ er } \Theta(\sqrt{n})$$

$$\log n^2 + n^{0.01} \text{ er } \Omega(2^n)$$

Opgave 2 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n .

Ja Nej

$$6\sqrt{n} \text{ er } O(n \cdot \log n)$$

$$3^3 \text{ er } O(\log n)$$

$$n^{2/3} \cdot n^{1/3} \text{ er } O(n \cdot \log n)$$

$$2(\log n)^2 \text{ er } O(2^n)$$

$$2^2 \text{ er } O(n \cdot \log n)$$

$$n \text{ er } O(n^{1/3})$$

$$n^{2/3} \text{ er } O(n)$$

$$2^{\log n} \text{ er } O(n^{2/3})$$

$$\sqrt{n} \text{ er } O(n^2)$$

$$8^{\log n} \text{ er } \Theta(n \cdot \log n)$$

$$\log n + n \cdot \log n \text{ er } \Omega(4^{\log n})$$

$$n^{2/3}/2 \text{ er } \Omega(n)$$

Opgave 3 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

- $(\log n)^3$ er $O(\sqrt{n})$
- $4 \cdot 8^{\log n}$ er $O(n^{0.01})$
- $5n^{1/3} + 2^2$ er $O(3^3)$
- 3 er $O(n^{0.001})$
- $n^{0.001} + n$ er $O(n)$
- 1 er $O(n^{1/3})$
- $4 \log(n!) + \sqrt{n}$ er $O(\log(n!))$
- $n^{0.01}$ er $O(2^{\log n})$
- $n^{0.01} + 5\sqrt{n} \cdot \log n$ er $O(n^{0.1})$
- 7 er $\Theta(5^5)$
- $4n^{2/3}$ er $\Theta(n)$
- $n \cdot \log n$ er $\Theta(1)$

Opgave 4 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

- $n^{3/2}$ er $O(n^{0.1})$
- $\sqrt{n} \cdot \log n + n^{1/3}/4$ er $O(1)$
- $\log n + \sqrt{n}$ er $O(n \cdot \log n)$
- $n \cdot \log n$ er $O(\sqrt{n})$
- $(\log n)^3 + \sum_{i=1}^n i$ er $O(\sqrt{n})$
- $\sqrt{n} \cdot \log n$ er $O((\log n)^3)$
- n er $O(2^{3 \log n})$
- $n^2 \log n + 8^{\log n}$ er $O(n^{0.001})$
- $n^{3/2}$ er $O(n)$
- $3n^{2/3} \cdot n^{1/3}$ er $\Theta(2^{3 \log n})$
- $5(\log n)^2$ er $\Omega(n \cdot \log n)$
- 2^n er $\Theta(n^3)$

Opgave 5 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

$$2 \cdot 8^{\log n} \text{ er } O(\log n^2)$$

$$1 \text{ er } O(n \cdot \log n)$$

$$\sqrt{n} \text{ er } O(n \cdot \log n)$$

$$2^{2 \log n} + 4^{\log n} \text{ er } O(\sqrt{n})$$

$$2 \log(n!) \text{ er } O(\sum_{i=1}^n i)$$

$$n^3 \text{ er } O((\log n)^7)$$

$$n! \text{ er } O(\sqrt{n} \cdot \log n)$$

$$7n^2 \text{ er } O(2^n)$$

$$n^{0.001} + 2^{\log n} \text{ er } O(n^{1/3})$$

$$n\sqrt{n} \text{ er } \Theta(n^{3/2})$$

$$\log n^2 \text{ er } \Theta(\log n)$$

$$8^{\log n} \text{ er } \Theta(2^{3 \log n})$$

Opgave 6 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

$$n^{0.01} \text{ er } O(\sqrt{n})$$

$$2^2 \text{ er } O((\log n)^3)$$

$$3n! + n^{2/3} \text{ er } O(\log n)$$

$$3n \cdot \log n + \log(n!) \text{ er } O((\log n)^2)$$

$$n^{2/3} \text{ er } O(n\sqrt{n})$$

$$2^{2 \log n} \text{ er } O(n^2)$$

$$n \text{ er } O(n^2 \log n)$$

$$(\log n)^6 \text{ er } O(2^n)$$

$$5n^2 \text{ er } O(\log n^2)$$

$$n \cdot \log n \text{ er } \Omega(n)$$

$$5n^{3/2} \text{ er } \Theta(n^{3/2})$$

$$n \cdot \log n \text{ er } \Omega(2^{3 \log n})$$

Opgave 7 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n .

Ja Nej

$$\log n + \log n^2 \text{ er } O(\log n^2)$$

$$n \cdot \log n \text{ er } O(\sqrt{n})$$

$$4n^3 + \sqrt{n} \cdot \log n \text{ er } O(\sqrt{n})$$

$$n \cdot \log n + n \text{ er } O(n^2 \log n)$$

$$n \cdot \log n \text{ er } O(n^n)$$

$$2^n \text{ er } O(2^{2 \log n})$$

$$n^2/7 \text{ er } O(n \cdot \log n)$$

$$n^{0.01} \text{ er } O(n^{3/2})$$

$$8^{\log n} \text{ er } O(\sqrt{n})$$

$$n^{2/3}/3 + \sqrt{n} \text{ er } \Omega(n^{0.1})$$

$$n^{2/3} \text{ er } \Theta(n^{2/3} \cdot n^{1/3})$$

$$n \text{ er } \Theta(2^{\log n})$$

Opgave 8 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n .

Ja Nej

$$6 \text{ er } O(\log n)$$

$$\sqrt{n} \text{ er } O(n^{0.1})$$

$$8^{\log n} \text{ er } O(2)$$

$$n \cdot \log n \text{ er } O(n\sqrt{n})$$

$$6 \cdot 3^n \text{ er } O(4^{\log n})$$

$$n^{0.01}/7 \text{ er } O(\log n)$$

$$\log n^2 \text{ er } O(n^{0.001})$$

$$\sqrt{n} + n^n \text{ er } O(n^{0.01})$$

$$n \text{ er } O((\log n)^2)$$

$$\log n^2 \text{ er } \Theta(\log n)$$

$$2(\log n)^3 + \sqrt{n} \text{ er } \Omega(\log(n!))$$

$$\log(n!) \text{ er } \Omega(n)$$

Opgave 9 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

$$n^{2/3} \cdot n^{1/3} \text{ er } O((\log n)^2)$$

$$\sqrt{n} \text{ er } O(n)$$

$$6n \cdot \log n + (\log n)^3 \text{ er } O((\log n)^4)$$

$$3 \log(n!) \text{ er } O((\log n)^3)$$

$$2^{2 \log n} \text{ er } O(n^{2/3})$$

$$n!/2 \text{ er } O(2^{\log n})$$

$$\sqrt{n} \cdot \log n \text{ er } O(8^{\log n})$$

$$n \cdot \log n \text{ er } O(\log n)$$

$$n \text{ er } O(\sqrt{n})$$

$$\sqrt{n} \cdot \log n \text{ er } \Omega(n \cdot \log n)$$

$$2^n + (\log n)^3 \text{ er } \Omega(n^{2/3} \cdot n^{1/3})$$

$$4^4 \text{ er } \Omega(n^{3/2})$$

Opgave 10 (Asymptotisk notation, 6 %)

I det følgende angiver log n 2-tals-logaritmen af n.

Ja Nej

$$\sqrt{n} \text{ er } O(n \cdot \log n)$$

$$(\log n)^5 + (\log n)^5 \text{ er } O(n^{3/2})$$

$$2 \log n \text{ er } O((\log n)^7)$$

$$\sqrt{n} \text{ er } O(n^{0.001})$$

$$2^{\log n} \text{ er } O(2^{3 \log n})$$

$$2^{3 \log n} \text{ er } O(8^{\log n})$$

$$n^{2/3} \text{ er } O(n^{0.1})$$

$$n \cdot \log n \text{ er } O(2^{3 \log n})$$

$$2^{3 \log n} \text{ er } O(n)$$

$$n \cdot \log n \text{ er } \Theta(\log(n!))$$

$$n^{0.1} \text{ er } \Omega(n^{1/3})$$

$$n^2 \text{ er } \Omega((\log n)^2)$$

Opgave 11 (Analyse af løkker, 6 %)**Algoritme** loop1(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop2(n)

```

 $s = 1$ 
for  $i = n$  to 1 step -1
    for  $j = n$  to 1 step -1
         $s = s + 1$ 

```

Algoritme loop3(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
    for  $j = 1$  to  $n$ 
         $s = s + 1$ 

```

Algoritme loop4(n)

```

 $i = 0$ 
 $s = 0$ 
while  $s \leq n$ 
     $i = i + 1$ 
     $s = s + i$ 

```

Algoritme loop5(n)

```

 $i = 1$ 
while  $i \leq n * n$ 
     $i = 2 * i$ 

```

Algoritme loop6(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
     $j = 1$ 
    while  $j \leq s$ 
         $j = j + 1$ 
     $s = 2 * s$ 

```

Angiv for hver af ovenstående algoritmer udførelstiden som funktion af n i Θ -notation.

$\Theta(\sqrt{n})$ $\Theta(n^2)$ $\Theta(\log n)$ $\Theta(n^3)$ $\Theta(n)$ $\Theta(n \log n)$ $\Theta(2^n)$ $\Theta(n\sqrt{n})$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 12 (Analyse af løkker, 6 %)**Algoritme** loop1(n)
 $i = 1$
while $i \leq n$
 $i = 3 * i$
Algoritme loop2(n)
 $s = 1$
for $i = 1$ **to** n
for $j = 1$ **to** n
 $s = s + 1$
Algoritme loop3(n)
 $i = 1$
while $i \leq n$
 $i = 2 * i$
Algoritme loop4(n)
 $i = 1$
while $i \leq n * n$
 $i = 3 * i$
Algoritme loop5(n)
 $i = 1$
while $i \leq n$
while $j \leq n$
 $j = j + 1$
 $i = 2 * i$
Algoritme loop6(n)
 $i = 1$
while $i \leq n$
 $j = i$
while $j \leq n$
 $j = j + 1$
 $i = 2 * i$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

 $\Theta(\log n)$ $\Theta(n)$ $\Theta((\log n)^2)$ $\Theta(n \log n)$ $\Theta(\sqrt{n})$ $\Theta(n\sqrt{n})$ $\Theta(n^3)$ $\Theta(n^2)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 13 (Analyse af løkker, 6 %)

Algoritme loop1(n)

```
i = 1
while i ≤ n
    i = i + i
```

Algoritme loop2(n)

```
s = 1
for i = 1 to n
    for j = 1 to n
        s = s + 1
```

Algoritme loop3(n)

```
s = 0
for i = 1 to n
    for j = 1 to n
        for k = 1 to n
            s = s + 1
```

Algoritme loop4(n)

```
s = 0
for i = 1 to n
    for j = i to n
        for k = i to j
            s = s + 1
```

Algoritme loop5(n)

```
i = 1
j = n
while i ≤ j
    i = i * 2
    j = ⌊j/2⌋
```

Algoritme loop6(n)

```
s = 0
i = n
while i > 1
    for j = 1 to n
        s = s + 1
        i = ⌊i/2⌋
```

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\log \log n)$ $\Theta(n^2)$ $\Theta(\sqrt{n})$ $\Theta(n)$ $\Theta(\log n)$ $\Theta(n^3)$ $\Theta((\log n)^2)$ $\Theta(n \log n)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 14 (Analyse af løkker, 6 %)**Algoritme** loop1(n)
 $i = 1$
while $i \leq n$
 $i = 2 * i$
Algoritme loop2(n)
 $i = 1$
while $i \leq n$
 $i = i + i$
Algoritme loop3(n)
 $s = 0$
for $i = 1$ **to** n
for $j = 1$ **to** n
for $k = 1$ **to** n
 $s = s + 1$
Algoritme loop4(n)
 $i = 1$
while $i \leq n * n$
 $i = 3 * i$
Algoritme loop5(n)
 $i = 1$
while $i \leq n$
 $j = 1$
while $j \leq i$
 $j = 2 * j$
 $i = i + 1$
Algoritme loop6(n)
 $i = n$
while $i > 0$
if i ulige
 $i = i - 1$
else
 $i = i/2$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\sqrt{n})$ $\Theta(n^2 \cdot \log n)$ $\Theta(\log n)$ $\Theta((\log n)^2)$ $\Theta(n)$ $\Theta(n^3)$ $\Theta(n^2)$ $\Theta(n \log n)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 15 (Analyse af løkker, 6 %)**Algoritme** loop1(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
  for  $j = i$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop2(n)

```

 $s = 1$ 
while  $s \leq n$ 
   $s = s + 1$ 

```

Algoritme loop3(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
  for  $j = i$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop4(n)

```

 $i = 1$ 
while  $i \leq n * n$ 
   $i = 3 * i$ 

```

Algoritme loop5(n)

```

 $i = 0$ 
while  $s \leq n$ 
   $i = i + 1$ 
   $s = s + i$ 

```

Algoritme loop6(n)

```

for  $i = 0$  to  $n$ 
   $j = 0$ 
   $s = 0$ 
  while  $s \leq i$ 
     $j = j + 1$ 
     $s = s + j$ 

```

Angiv for hver af ovenstående algoritmer udførelstiden som funktion af n i Θ -notation.

$\Theta(\sqrt{n})$ $\Theta(n)$ $\Theta(n \log n)$ $\Theta(\log n)$ $\Theta(\log \log n)$ $\Theta(n^2)$ $\Theta(n\sqrt{n})$ $\Theta(n^3)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 16 (Analyse af løkker, 6 %)**Algoritme** loop1(n)
 $s = 1$
for $i = 1$ **to** n
 for $j = i$ **to** n
 $s = s + 1$
Algoritme loop2(n)
 $i = 1$
while $i \leq n$
 $i = i + i$
Algoritme loop3(n)
 $i = 1$
while $i \leq n$
 $j = 1$
 while $j \leq n$
 $j = j + 1$
 $i = i + 1$
Algoritme loop4(n)
for $i = 1$ **to** n
 $j = 1$
 while $j \leq n$
 $j = 2 * j$
Algoritme loop5(n)
 $i = 1$
while $i \leq n * n$
 $i = 2 * i$
Algoritme loop6(n)
 $s = 0$
 $j = 0$
for $i = 1$ **to** n
 $j = j + i$
 for $k = 1$ **to** j
 $s = s + 1$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(n^3)$ $\Theta(\log n)$ $\Theta(\log \log n)$ $\Theta(\sqrt{n})$ $\Theta(\sqrt[3]{n})$ $\Theta(n)$ $\Theta(n \log n)$ $\Theta(n^2)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 17 (Analyse af løkker, 6 %)

Algoritme loop1(n)
 $i = 1$
while $i \leq n$
 $j = 1$
while $j \leq n$
 $j = j + 1$
 $i = i + 1$

Algoritme loop2(n)
 $s = 1$
for $i = n$ **to** 1 **step** -1
for $j = n$ **to** 1 **step** -1
 $s = s + 1$

Algoritme loop3(n)
for $i = 1$ **to** n
 $j = i$
while $j > 0$
 $j = j - 1$

Algoritme loop4(n)
 $i = 1$
while $i \leq n * n$
 $i = 3 * i$

Algoritme loop5(n)
 $s = 1$
for $i = 1$ **to** n
for $j = 1$ **to** n
 $s = s + 1$
for $k = 1$ **to** n
 $s = s + 1$

Algoritme loop6(n)
for $i = 1$ **to** n
 $j = 1$
while $j \leq i$
 $j = 2 * j$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(n \log n)$ $\Theta(n^2)$ $\Theta(n^3)$ $\Theta((\log n)^2)$ $\Theta(\sqrt{n})$ $\Theta(\log n)$ $\Theta(n)$ $\Theta(2^n)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 18 (Analyse af løkker, 6 %)**Algoritme** loop1(n)
 $i = 1$
while $i \leq n$
 $i = 3 * i$
 $s = 1$
for $i = 1$ **to** n
 $s = s + 1$
Algoritme loop3(n)
 $s = 1$
for $i = 1$ **to** n
for $j = 1$ **to** n
 $s = s + 1$
Algoritme loop4(n)
 $i = 1$
while $i \leq n$
 $j = 0$
while $j \leq n$
 $j = j + 1$
 $i = 2 * i$
 $i = 1$
 $j = n$
while $i \leq j$
 $i = i * 2$
 $j = \lfloor j/2 \rfloor$
Algoritme loop6(n)
 $i = 1$
while $i \leq n$
 $j = 1$
while $j \leq i$
 $j = j + 1$
 $i = 2 * i$

Angiv for hver af ovenstående algoritmer udførelstiden som funktion af n i Θ -notation.

 $\Theta\left(\frac{\log n}{\log \log n}\right) \quad \Theta(n\sqrt{n}) \quad \Theta(n^2) \quad \Theta(\log n) \quad \Theta(n) \quad \Theta(\sqrt{n}) \quad \Theta(n \log n) \quad \Theta(n^3)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 19 (Analyse af løkker, 6 %)**Algoritme** loop1(n)
 $i = 1$
while $i \leq n$
 $i = 2 * i$
Algoritme loop2(n)
 $s = 1$
for $i = 1$ **to** n
 $s = s + 1$
Algoritme loop3(n)
 $i = n$
while $i > 0$
 $i = i - 1$
Algoritme loop4(n)
 $s = 0$
for $i = 1$ **to** n
for $j = i$ **to** n
for $k = i$ **to** j
 $s = s + 1$
Algoritme loop5(n)
 $i = 1$
while $i \leq n * n$
 $i = 3 * i$
Algoritme loop6(n)
 $i = 1$
 $j = 1$
 $s = 0$
while $s \leq n$
while $j \leq s$
 $j = 2 * j$
 $s = s + i$
 $i = i + 1$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

 $\Theta(n \log n)$ $\Theta(\sqrt{n})$ $\Theta(n^3)$ $\Theta(n^2)$ $\Theta(n)$ $\Theta((\log n)^2)$ $\Theta(2^n)$ $\Theta(\log n)$

loop1

loop2

loop3

loop4

loop5

loop6

Opgave 20 (Analyse af løkker, 6 %)**Algoritme** loop1(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $i * i$ 
     $s = s + 1$ 

```

Algoritme loop2(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $n$ 
    for  $k = 1$  to  $n$ 
       $s = s + 1$ 

```

Algoritme loop3(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
   $s = s + 1$ 

```

Algoritme loop4(n)

```

 $i = 1$ 
while  $i \leq n$ 
   $j = 1$ 
  while  $j \leq i$ 
     $j = j + 1$ 
   $i = i + 1$ 

```

Algoritme loop5(n)

```

 $i = 1$ 
while  $i \leq n$ 
   $j = 1$ 
  while  $j \leq i$ 
     $j = 2 * j$ 
   $i = i + 1$ 

```

Algoritme loop6(n)

```

 $i = 1$ 
while  $i \leq n$ 
   $j = i$ 
  while  $j \leq n$ 
     $j = 2 * j$ 
   $i = 2 * i$ 

```

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\log n)$ $\Theta(\sqrt{n})$ $\Theta(n^3)$ $\Theta(n)$ $\Theta(n\sqrt{n})$ $\Theta(n \log n)$ $\Theta((\log n)^2)$ $\Theta(n^2)$

loop1

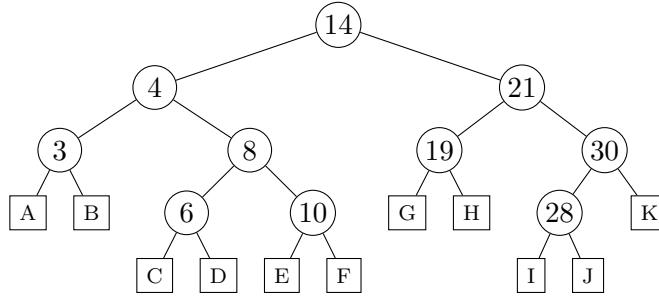
loop2

loop3

loop4

loop5

loop6

Opgave 21 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancede binære søgetræe elementerne 24, 2, 25, 15 og 5 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

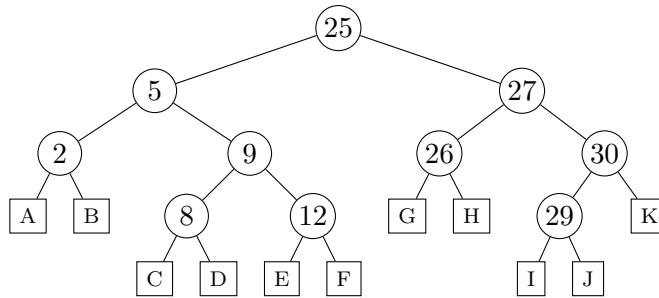
INSERT(24)

INSERT(2)

INSERT(25)

INSERT(15)

INSERT(5)

Opgave 22 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancede binære søgetræe elementerne 1, 31, 14, 13 og 23 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

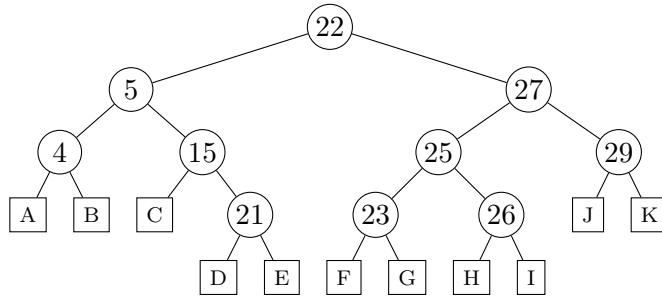
INSERT(1)

INSERT(31)

INSERT(14)

INSERT(13)

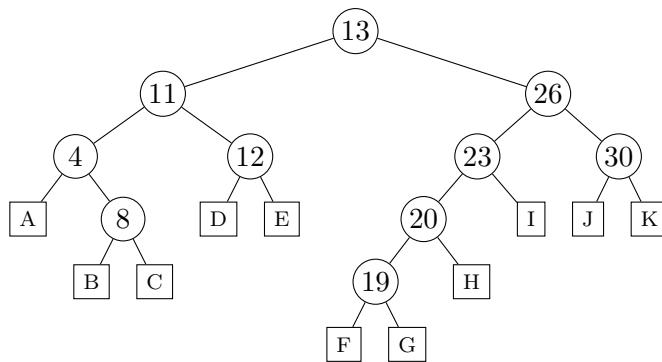
INSERT(23)

Opgave 23 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancerede binære søgetræe elementerne 20, 7, 9, 28 og 14 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A	B	C	D	E	F	G	H	I	J	K
---	---	---	---	---	---	---	---	---	---	---

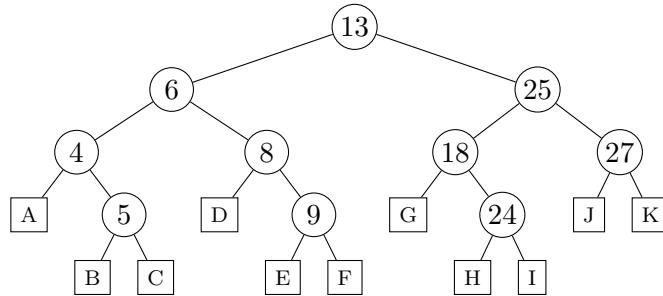
INSERT(20)
 INSERT(7)
 INSERT(9)
 INSERT(28)
 INSERT(14)

Opgave 24 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancerede binære søgetræe elementerne 14, 25, 7, 10 og 6 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A	B	C	D	E	F	G	H	I	J	K
---	---	---	---	---	---	---	---	---	---	---

INSERT(14)
 INSERT(25)
 INSERT(7)
 INSERT(10)
 INSERT(6)

Opgave 25 (Indsættelser i søgeræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancede binære søgeræe elementerne 19, 28, 21, 12 og 16 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

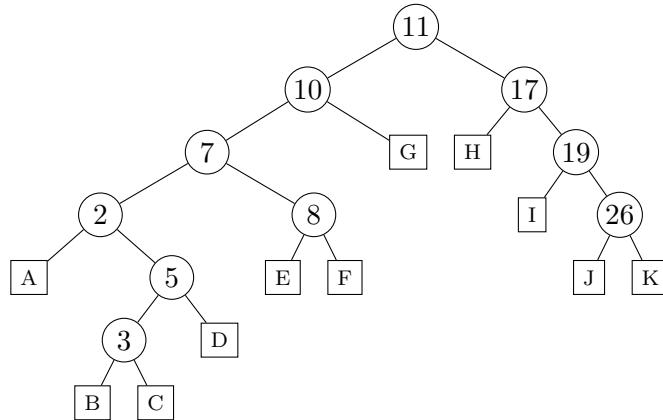
INSERT(19)

INSERT(28)

INSERT(21)

INSERT(12)

INSERT(16)

Opgave 26 (Indsættelser i søgeræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancerede binære søgeræe elementerne 24, 13, 20, 1 og 4 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

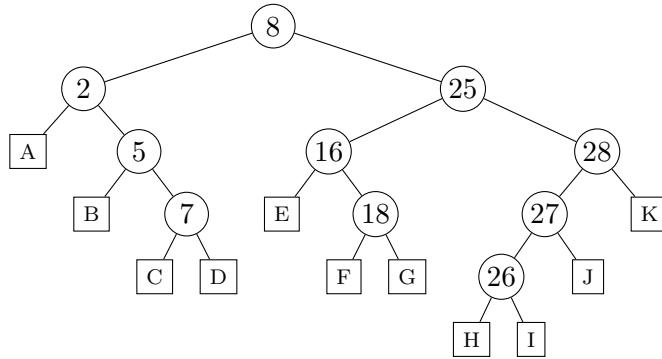
INSERT(24)

INSERT(13)

INSERT(20)

INSERT(1)

INSERT(4)

Opgave 27 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancerede binære søgetræe elementerne 24, 4, 1, 13 og 23 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

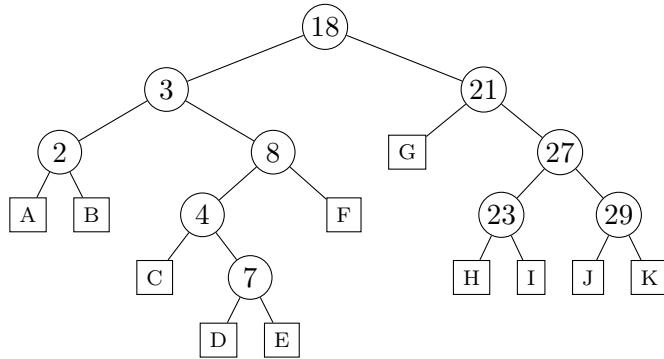
INSERT(24)

INSERT(4)

INSERT(1)

INSERT(13)

INSERT(23)

Opgave 28 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancede binære søgetræe elementerne 24, 14, 30, 20 og 19 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

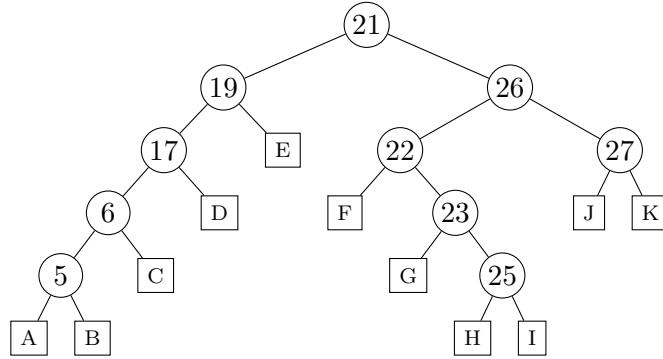
INSERT(24)

INSERT(14)

INSERT(30)

INSERT(20)

INSERT(19)

Opgave 29 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancerede binære søgetræe elementerne 24, 18, 14, 4 og 7 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

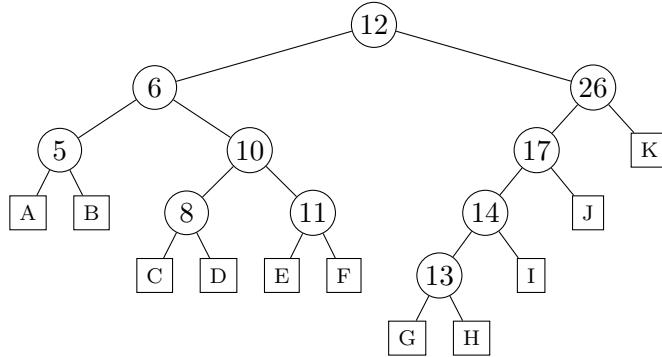
INSERT(24)

INSERT(18)

INSERT(14)

INSERT(4)

INSERT(7)

Opgave 30 (Indsættelser i søgetræer, 4 %)

Angiv i hvilke blade A–K i ovenstående ubalancede binære søgetræe elementerne 7, 15, 22, 16 og 18 skal indsættes (det antages at før hver indsættelse indeholder træet kun ovenstående ti elementer).

A B C D E F G H I J K

INSERT(7)

INSERT(15)

INSERT(22)

INSERT(16)

INSERT(18)

Opgave 31 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 1, 8, 2, 11, 7, 3 og 13 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
1	8	2	11	7	3	13

1	2	3	4	5	6	7
13	8	11	1	7	2	3

1	2	3	4	5	6	7
13	11	3	8	7	1	2

1	2	3	4	5	6	7
8	11	13	1	7	3	2

1	2	3	4	5	6	7
13	11	8	7	3	2	1

Opgave 32 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 10, 7, 12, 14, 2, 13 og 9 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
10	7	12	14	2	13	9
12	14	13	7	2	10	9
14	12	13	7	2	10	9
14	10	13	7	2	12	9
14	13	12	10	9	7	2

Opgave 33 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 10, 5, 7, 2, 14, 9 og 11 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
14	10	11	2	5	9	7
10	5	7	2	14	9	11
14	11	10	9	7	5	2
14	10	11	2	5	7	9
10	14	11	2	5	9	7

Opgave 34 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 7, 8, 1, 4, 14, 5 og 10 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
14	8	10	4	7	5	1
7	8	1	4	14	5	10
14	10	8	7	5	4	1
8	14	10	4	7	5	1
14	8	10	4	7	1	5

Opgave 35 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 2, 5, 8, 1, 11, 4 og 7 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
11	5	8	1	2	4	7
1	2	3	4	5	6	7
2	5	8	1	11	4	7
1	2	3	4	5	6	7
11	8	7	5	4	2	1
1	2	3	4	5	6	7
11	8	7	1	2	4	5
1	2	3	4	5	6	7
8	11	7	1	5	4	2

Opgave 36 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 4, 11, 6, 3, 9, 14 og 2 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
14	9	11	3	4	6	2
1	2	3	4	5	6	7
4	11	6	3	9	14	2
1	2	3	4	5	6	7
11	9	14	3	4	6	2
1	2	3	4	5	6	7
14	11	9	6	4	3	2
1	2	3	4	5	6	7
14	11	6	3	9	4	2

Opgave 37 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 7, 3, 9, 13, 10, 6 og 11 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
13	10	11	3	9	6	7
1	2	3	4	5	6	7
13	11	10	9	7	6	3
1	2	3	4	5	6	7
13	10	11	3	7	6	9
1	2	3	4	5	6	7
9	13	11	3	10	6	7
1	2	3	4	5	6	7
7	3	9	13	10	6	11

Opgave 38 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 9, 7, 3, 12, 14, 5 og 13 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
14	12	13	9	7	5	3
9	7	3	12	14	5	13
14	13	12	9	7	5	3
14	12	13	7	9	3	5
9	14	13	12	7	5	3

Opgave 39 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 2, 1, 13, 14, 4, 8 og 7 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
14	13	8	7	4	2	1
14	13	8	1	4	2	7
2	1	13	14	4	8	7
13	14	8	1	4	2	7
14	4	13	1	2	8	7

Opgave 40 (Max-Heap-Insert, 4 %)

Angiv den binære max-heap efter indsættelse af elementerne 6, 2, 8, 13, 12, 10 og 5 i den givne rækkefølge med MAX-HEAP-INSERT, startende med den tomme heap.

1	2	3	4	5	6	7
13	12	10	2	8	6	5
6	2	8	13	12	10	5
13	12	10	2	6	8	5
8	13	10	2	12	6	5
13	12	10	8	6	5	2

Opgave 41 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

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

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

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

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

Opgave 42 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

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

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

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

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

Opgave 43 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 44 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 45 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 46 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 47 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 48 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

Opgave 49 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

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

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

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

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

Opgave 50 (Build-Max-Heap, 4 %)

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

Hvad er resultatet af BUILD-MAX-HEAP på ovenstående array ?

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

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

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

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

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

Opgave 51 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
26	25	23	16	15	22	9	5	11	7	3	2	18

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12
25	16	23	11	15	22	9	5	7	3	2	18
1	2	3	4	5	6	7	8	9	10	11	12
25	18	23	16	15	22	9	5	11	7	3	2
1	2	3	4	5	6	7	8	9	10	11	13
25	16	23	11	15	22	9	5		7	3	2
1	2	3	4	5	6	7	8	9	10	11	12
25	16	23	11	15	22	9	5	18	7	3	2
1	2	3	4	5	6	7	8	9	10	11	12
25	23	18	15	22	16	5	11	7	3	2	9

Opgave 52 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
26	25	22	12	17	21	18	9	3	6	8	15	14

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12
25	17	22	12	8	21	18	9	3	6	14	15
1	2	3	4	5	6	7	8	9	10	11	12
25	17	22	12	14	21	18	9	3	6	8	15
1	2	3	4	5	6	7	8	9	10	11	12
25	17	22	12	8	21	18	9	3	6	15	14
1	2	3	4	5	6	7	8	9	10	11	13
25	17	22	12	8	21	18	9	3	6		15
1	2	3	4	5	6	7	8	9	10	11	12
25	22	18	17	21	14	9	3	6	8	15	12

Opgave 53 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
24	23	21	16	20	17	2	12	8	13	4	5	15

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	
23	20	21	16	15	17	2	12	8	13	4	5	
1	2	3	4	5	6	7	8	9	10	11	12	
23	20	21	16	13	17	2	12	8		4	5	15
1	2	3	4	5	6	7	8	9	10	11	12	
23	21	16	20	17	15	12	8	13	4	5	2	
1	2	3	4	5	6	7	8	9	10	11	12	
23	20	21	16	13	17	2	12	8	4	5	15	
1	2	3	4	5	6	7	8	9	10	11	12	
23	20	21	16	13	17	2	12	8	15	4	5	

Opgave 54 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
26	22	21	19	13	20	8	6	11	5	10	1	12

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	
22	19	21	11	13	20	8	6	5	10	1	12	
1	2	3	4	5	6	7	8	9	10	11	12	
22	19	21	12	13	20	8	6	11	5	10	1	
1	2	3	4	5	6	7	8	9	10	11	12	
22	19	21	11	13	20	8	6		5	10	1	12
1	2	3	4	5	6	7	8	9	10	11	12	
22	19	21	11	13	20	8	6	12	5	10	1	
1	2	3	4	5	6	7	8	9	10	11	12	
22	21	19	13	20	12	6	11	5	10	1	8	

Opgave 55 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
26	25	21	23	18	20	14	10	3	4	9	12	11

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	13
25	23	21	10	18	20	14		3	4	9	12	11
1	2	3	4	5	6	7	8	9	10	11	12	
25	21	23	18	20	14	10	3	4	9	12	11	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	21	10	18	20	14	3	4	9	12	11	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	21	11	18	20	14	10	3	4	9	12	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	21	10	18	20	14	11	3	4	9	12	

Opgave 56 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
25	24	20	11	4	16	6	7	10	2	1	13	14

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	
24	14	20	11	4	16	6	7	10	2	1	13	
1	2	3	4	5	6	7	8	9	10	11	12	
24	11	20	10	4	16	6	7	14	2	1	13	
1	2	3	4	5	6	7	8	9	10	11	12	
24	11	20	10	4	16	6	7	2	1	13	14	
1	2	3	4	5	6	7	8	9	10	11	12	
24	11	20	10	4	16	6	7		2	1	13	14
1	2	3	4	5	6	7	8	9	10	11	12	
24	20	14	10	16	11	7	4	2	1	13	6	

Opgave 57 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
25	24	23	20	19	22	18	9	3	16	12	2	10

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12
24	20	23	9	19	22	18	3	16	12	2	10
1	2	3	4	5	6	7	8	9	10	11	12
24	23	20	19	22	18	9	3	16	12	2	10
1	2	3	4	5	6	7	8	9	10	11	12
24	20	23	9	19	22	18	10	3	16	12	2
1	2	3	4	5	6	7	8	9	10	11	12
24	20	23	9	19	22	18		3	16	12	2
1	2	3	4	5	6	7	8	9	10	11	12
24	20	23	10	19	22	18	9	3	16	12	2

Opgave 58 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
25	24	21	23	11	17	9	1	4	2	5	6	8

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12
24	23	21	4	11	17	9	1	2	5	6	8
1	2	3	4	5	6	7	8	9	10	11	12
24	23	21	4	11	17	9	1	8	2	5	6
1	2	3	4	5	6	7	8	9	10	11	12
24	21	23	11	17	9	1	4	2	5	6	8
1	2	3	4	5	6	7	8	9	10	11	12
24	23	21	4	11	17	9	1		2	5	6
1	2	3	4	5	6	7	8	9	10	11	12
24	23	21	8	11	17	9	1	4	2	5	6

Opgave 59 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
25	21	19	18	11	17	16	14	1	4	9	7	15

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	
21	19	18	11	17	16	14	1	4	9	7	15	
1	2	3	4	5	6	7	8	9	10	11	12	
21	18	19	14	11	17	16	15	1	4	9	7	
1	2	3	4	5	6	7	8	9	10	11	12	
21	18	19	15	11	17	16	14	1	4	9	7	
1	2	3	4	5	6	7	8	9	10	11	12	
21	18	19	14	11	17	16	1	4	9	7	15	
1	2	3	4	5	6	7	8	9	10	11	12	13
21	18	19	14	11	17	16		1	4	9	7	15

Opgave 60 (Heap-Extract-Max, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13
26	25	22	17	23	20	6	5	9	10		13	18

Hvad er resultatet af HEAP-EXTRACT-MAX på ovenstående max-heap ?

1	2	3	4	5	6	7	8	9	10	11	12	13
25	23	22	17	15	20	6	5	9	10		13	18
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	22	17	18	20	6	5	9	10	15	13	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	22	17	15	20	6	5	9	10	18	13	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	18	22	20	17	5	9	10	15	13	6	
1	2	3	4	5	6	7	8	9	10	11	12	
25	23	22	17	15	20	6	5	9	10	13	18	

Opgave 61 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9	20	29	8	23	15	18	24	30	5	10	11	13	17	28

Angiv resultatet af at anvende $\text{PARTITION}(A, 3, 12)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9	20	8	5	10	11	18	24	30	29	23	15	13	17	28
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
5	8	9	10	11	13	15	17	18	20	23	24	28	29	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9	20	5	8	10	11	15	18	23	24	29	30	13	17	28
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9	20	8	5	10	11	29	23	15	18	24	30	13	17	28

Opgave 62 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	18	28	11	14	15	12	22	30	7	29	9	10	21	3

Angiv resultatet af at anvende $\text{PARTITION}(A, 3, 13)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	18	7	9	10	15	12	22	30	28	29	11	14	21	3
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	7	9	10	11	12	14	15	18	21	22	27	28	29	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	18	7	9	10	11	12	14	15	22	28	29	30	21	3
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	18	7	9	10	28	11	14	15	12	22	30	29	21	3

Opgave 63 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
24	6	15	12	9	5	17	7	1	18	30	20	26	29	14

Angiv resultatet af at anvende $\text{PARTITION}(A, 4, 13)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
24	6	15	12	9	5	17	7	1	18	30	20	26	29	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	5	6	7	9	12	14	15	17	18	20	24	26	29	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
24	6	15	1	5	7	9	12	17	18	20	26	30	29	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
24	6	15	12	9	5	17	7	1	18	20	26	30	29	14

Opgave 64 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
30	25	18	2	3	20	27	11	21	9	28	12	7	23	6

Angiv resultatet af at anvende $\text{PARTITION}(A, 4, 13)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
30	25	18	2	3	7	27	11	21	9	28	12	20	23	6
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2	3	6	7	9	11	12	18	20	21	23	25	27	28	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
30	25	18	2	3	7	9	11	12	20	21	27	28	23	6
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
30	25	18	2	3	7	20	27	11	21	9	28	12	23	6

Opgave 65 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	19	5	1	17	15	22	14	7	29	28	9	6	26	11

Angiv resultatet af at anvende $\text{PARTITION}(A, 3, 12)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	19	5	1	7	9	22	14	17	29	28	15	6	26	11
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	3	5	6	7	9	11	14	15	17	19	22	26	28	29
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	19	1	5	7	9	14	15	17	22	28	29	6	26	11
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
3	19	5	1	7	9	17	15	22	14	29	28	6	26	11

Opgave 66 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	26	14	6	27	15	12	19	28	16	17	22	25	18	4

Angiv resultatet af at anvende $\text{PARTITION}(A, 2, 14)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	14	6	15	12	16	17	18	28	26	27	22	25	19	4
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4	6	11	12	14	15	16	17	18	19	22	25	26	27	28
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	6	12	14	15	16	17	18	19	22	25	26	27	28	4
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	14	6	15	12	16	17	18	26	27	19	28	22	25	4

Opgave 67 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
28	2	4	23	30	25	7	17	24	16	1	11	19	13	14

Angiv resultatet af at anvende $\text{PARTITION}(A, 4, 12)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
28	2	4	7	1	11	23	17	24	16	30	25	19	13	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	2	4	7	11	13	14	16	17	19	23	24	25	28	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
28	2	4	1	7	11	16	17	23	24	25	30	19	13	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
28	2	4	7	1	11	23	30	25	17	24	16	19	13	14

Opgave 68 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	9	25	12	15	26	18	5	29	13	19	4	16	17	7

Angiv resultatet af at anvende $\text{PARTITION}(A, 4, 14)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	9	25	12	15	5	13	4	16	17	19	26	29	18	7
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4	5	7	9	12	13	15	16	17	18	19	25	26	27	29
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	9	25	4	5	12	13	15	16	17	18	19	26	29	7
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
27	9	25	12	15	5	13	4	16	17	26	18	29	19	7

Opgave 69 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
6	23	12	22	10	24	26	5	19	8	3	18	21	2	1

Angiv resultatet af at anvende $\text{PARTITION}(A, 3, 12)$ på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----

Opgave 70 (Partition, 4 %)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	30	29	6	21	26	8	4	23	1	7	15	9	19	13

Angiv resultatet af at anvende PARTITION($A, 2, 14$) på ovenstående array A .

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	6	8	4	1	7	15	9	19	21	26	29	30	23	13
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	4	6	7	8	9	11	13	15	19	21	23	26	29	30
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	1	4	6	7	8	9	15	19	21	23	26	29	30	13
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	6	8	4	1	7	15	9	19	30	29	21	26	23	13

Opgave 71 (Radix-sort, 4 %)

4142 2302 2402 2242 2202 4011

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

2202 2302 2402 4011 2242 4142
 2202 2242 2302 2402 4011 4142
 4011 2302 2402 2202 4142 2242
 2242 2202 2302 2402 4011 4142
 2302 2402 2202 4011 4142 2242

Opgave 72 (Radix-sort, 4 %)

3123 2443 2414 2023 1214 3243

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

1214 2023 2443 2414 3123 3243
 1214 2414 2023 3123 2443 3243
 2414 1214 3123 2023 2443 3243
 3123 2023 2443 3243 2414 1214
 1214 2023 2414 2443 3123 3243

Opgave 73 (Radix-sort, 4 %)

3403 4132 4342 0432 4103 0132

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

4132	0432	0132	4342	3403	4103
3403	4103	4132	0432	0132	4342
3403	4103	0132	0432	4132	4342
0132	0432	3403	4103	4132	4342
0132	0432	3403	4132	4103	4342

Opgave 74 (Radix-sort, 4 %)

1110 1010 2011 4311 2001 1101

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

1101	2001	1010	1110	2011	4311
1010	1101	1110	2001	2011	4311
1010	1110	1101	2011	2001	4311
2001	1101	1110	1010	2011	4311
1110	1010	2001	1101	2011	4311

Opgave 75 (Radix-sort, 4 %)

0134 2430 0130 2412 1212 3434

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

0130	0134	1212	2412	2430	3434
1212	2412	0130	2430	0134	3434
0134	0130	1212	2430	2412	3434
2430	0130	2412	1212	0134	3434
2412	1212	2430	0130	0134	3434

Opgave 76 (Radix-sort, 4 %)

0313 2413 2411 4102 3213 1413

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

4102	2411	0313	2413	3213	1413
0313	1413	2411	2413	3213	4102
4102	2411	0313	1413	2413	3213
2411	4102	0313	2413	3213	1413
0313	1413	2413	2411	3213	4102

Opgave 77 (Radix-sort, 4 %)

4311 4432 0032 0311 0120 0020

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

4311	0311	0120	0020	4432	0032
0120	0020	4311	0311	4432	0032
0032	0020	0120	0311	4311	4432
0020	0032	0120	0311	4311	4432
0311	4311	0020	0120	0032	4432

Opgave 78 (Radix-sort, 4 %)

1241 1301 4441 4141 0224 4124

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

0224	1241	1301	4124	4141	4441
1301	0224	4124	1241	4441	4141
1301	0224	4124	1241	4141	4441
0224	1241	1301	4141	4124	4441
1301	1241	4441	4141	0224	4124

Opgave 79 (Radix-sort, 4 %)

1031 0234 2031 0231 0034 2222

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

2222	1031	2031	0231	0234	0034
2222	0231	1031	2031	0034	0234
0034	0234	0231	1031	2031	2222
0034	0231	0234	1031	2031	2222
1031	2031	0231	2222	0234	0034

Opgave 80 (Radix-sort, 4 %)

4321 4430 0133 0130 0421 2321

Betrægt RADIX-SORT anvendt på ovenstående liste af tal ($d = 4, k = 5$). Angiv den delvist sortede liste efter at RADIX-SORT har sorteret tallene efter de *to* mindst betydende cifre.

0421	2321	4321	0130	4430	0133
0133	0130	0421	2321	4321	4430
4321	0421	2321	4430	0130	0133
0130	0133	0421	2321	4321	4430
4430	0130	4321	0421	2321	0133

Opgave 81 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
18				5	20				14	7

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 3k \bmod 11$.

Angiv positionerne de fem elementer 1, 3, 4, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 5, 7, 14, 18 og 20).

0	1	2	3	4	5	6	7	8	9	10
INSERT(1)										
	INSERT(3)									
		INSERT(4)								
			INSERT(9)							
				INSERT(10)						

Opgave 82 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
22	9							6	4	15

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 5k \bmod 11$.

Angiv positionerne de fem elementer 2, 5, 7, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 4, 6, 9, 15 og 22).

0	1	2	3	4	5	6	7	8	9	10
INSERT(2)										
	INSERT(5)									
		INSERT(7)								
			INSERT(10)							
				INSERT(11)						

Opgave 83 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11			16		12				15	4

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 5k \bmod 11$.

Angiv positionerne de fem elementer 2, 3, 5, 6 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 4, 11, 12, 15 og 16).

0 1 2 3 4 5 6 7 8 9 10

INSERT(2)

INSERT(3)

INSERT(5)

INSERT(6)

INSERT(10)

Opgave 84 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
					9		17	21	6	18

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 3k \bmod 11$.

Angiv positionerne de fem elementer 1, 3, 5, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 6, 9, 17, 18 og 21).

0 1 2 3 4 5 6 7 8 9 10

INSERT(1)

INSERT(3)

INSERT(5)

INSERT(10)

INSERT(11)

Opgave 85 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0			7	2	13					16

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 2k \bmod 11$.

Angiv positionerne de fem elementer 1, 3, 5, 6 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 2, 7, 13 og 16).

0	1	2	3	4	5	6	7	8	9	10
INSERT(1)										
	INSERT(3)									
		INSERT(5)								
			INSERT(6)							
				INSERT(11)						

Opgave 86 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11	22		18				9			16

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 2k \bmod 11$.

Angiv positionerne de fem elementer 0, 1, 2, 5 og 8 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 9, 11, 16, 18 og 22).

0	1	2	3	4	5	6	7	8	9	10
INSERT(0)										
	INSERT(1)									
		INSERT(2)								
			INSERT(5)							
				INSERT(8)						

Opgave 87 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11	22			14			19			13

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 5k \bmod 11$.

Angiv positionerne de fem elementer 0, 2, 4, 6 og 9 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 11, 13, 14, 19 og 22).

0	1	2	3	4	5	6	7	8	9	10
INSERT(0)										
	INSERT(2)									
		INSERT(4)								
			INSERT(6)							
				INSERT(9)						

Opgave 88 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
						7	21	10	5	19

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 4k \bmod 11$.

Angiv positionerne de fem elementer 0, 1, 2, 6 og 8 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 5, 7, 10, 19 og 21).

0	1	2	3	4	5	6	7	8	9	10
INSERT(0)										
	INSERT(1)									
		INSERT(2)								
			INSERT(6)							
				INSERT(8)						

Opgave 89 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
		12			14		4	21	15	

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 2k \bmod 11$.

Angiv positionerne de fem elementer 1, 3, 7, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 4, 12, 14, 15 og 21).

0 1 2 3 4 5 6 7 8 9 10

INSERT(1)

INSERT(3)

INSERT(7)

INSERT(10)

INSERT(11)

Opgave 90 (Lineær probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0	15						6	4	13	

I ovenstående hashtabel af størrelse 11 er anvendt *linear probing* med hashfunktionen $h(k) = 5k \bmod 11$.

Angiv positionerne de fem elementer 1, 2, 5, 8 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 4, 6, 13 og 15).

0 1 2 3 4 5 6 7 8 9 10

INSERT(1)

INSERT(2)

INSERT(5)

INSERT(8)

INSERT(11)

Opgave 91 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0		1			8	3			19	

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 2k \text{ mod } 11$ og $h = (h'(k) + i + 3i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 2, 4, 7, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 1, 3, 8 og 19).

0 1 2 3 4 5 6 7 8 9 10

INSERT(2)

INSERT(4)

INSERT(7)

INSERT(10)

INSERT(11)

Opgave 92 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11							21	13	16	5

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 4k \text{ mod } 11$ og $h = (h'(k) + i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 3, 4, 8, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 5, 11, 13, 16 og 21).

0 1 2 3 4 5 6 7 8 9 10

INSERT(3)

INSERT(4)

INSERT(8)

INSERT(9)

INSERT(10)

Opgave 93 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
22		7				21		18	15	

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 5k \text{ mod } 11$ og $h = (h'(k) + i + 5i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 2, 6, 8, 9 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 7, 15, 18, 21 og 22).

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

INSERT(2)
 INSERT(6)
 INSERT(8)
 INSERT(9)
 INSERT(11)

Opgave 94 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11		0					9	4		16

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 2k \text{ mod } 11$ og $h = (h'(k) + i + i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 1, 2, 5, 6 og 8 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 4, 9, 11 og 16).

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

INSERT(1)
 INSERT(2)
 INSERT(5)
 INSERT(6)
 INSERT(8)

Opgave 95 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0		7		1		18			6	

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 4k \bmod 11$ og $h = (h'(k) + 3i + 4i^2) \bmod 11$.

Angiv positionerne de fem elementer 2, 5, 9, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 1, 6, 7 og 18).

0 1 2 3 4 5 6 7 8 9 10

INSERT(2)

INSERT(5)

INSERT(9)

INSERT(10)

INSERT(11)

Opgave 96 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
		7	18	13			9	15		

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 2k \bmod 11$ og $h = (h'(k) + 3i + i^2) \bmod 11$.

Angiv positionerne de fem elementer 2, 4, 5, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 7, 9, 13, 15 og 18).

0 1 2 3 4 5 6 7 8 9 10

INSERT(2)

INSERT(4)

INSERT(5)

INSERT(10)

INSERT(11)

Opgave 97 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
			9			18		20	16	8

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 4k \bmod 11$ og $h = (h'(k) + 5i^2) \bmod 11$.

Angiv positionerne de fem elementer 3, 5, 7, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 8, 9, 16, 18 og 20).

0 1 2 3 4 5 6 7 8 9 10

INSERT(3)

INSERT(5)

INSERT(7)

INSERT(10)

INSERT(11)

Opgave 98 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
22	14					3	18	2		

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 4k \bmod 11$ og $h = (h'(k) + 2i + 3i^2) \bmod 11$.

Angiv positionerne de fem elementer 0, 5, 7, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 2, 3, 14, 18 og 22).

0 1 2 3 4 5 6 7 8 9 10

INSERT(0)

INSERT(5)

INSERT(7)

INSERT(9)

INSERT(10)

Opgave 99 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
	17	12	7	13						6

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 2k \text{ mod } 11$ og $h = (h'(k) + 4i + 5i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 2, 4, 5, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 6, 7, 12, 13 og 17).

0 1 2 3 4 5 6 7 8 9 10

INSERT(2)

INSERT(4)

INSERT(5)

INSERT(9)

INSERT(10)

Opgave 100 (Kvadratisk probing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
22		18					7	6	17	

I ovenstående hashtabel af størrelse 11 er anvendt *kvadratisk probing* med hashfunktionerne $h'(k) = 5k \text{ mod } 11$ og $h = (h'(k) + 2i + 3i^2) \text{ mod } 11$.

Angiv positionerne de fem elementer 0, 2, 4, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 6, 7, 17, 18 og 22).

0 1 2 3 4 5 6 7 8 9 10

INSERT(0)

INSERT(2)

INSERT(4)

INSERT(9)

INSERT(10)

Opgave 101 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
22					15	7	10			21

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 4k \text{ mod } 11$ og $h_2 = 1 + (2k \text{ mod } 10)$.

Angiv positionerne de fem elementer 1, 3, 4, 6 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 7, 10, 15, 21 og 22).

0	1	2	3	4	5	6	7	8	9	10
INSERT(1)										
	INSERT(3)									
		INSERT(4)								
			INSERT(6)							
				INSERT(11)						

Opgave 102 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
		12			8		9	16		5

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 2k \text{ mod } 11$ og $h_2 = 1 + (3k \text{ mod } 10)$.

Angiv positionerne de fem elementer 1, 2, 4, 6 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 5, 8, 9, 12 og 16).

0	1	2	3	4	5	6	7	8	9	10
INSERT(1)										
	INSERT(2)									
		INSERT(4)								
			INSERT(6)							
				INSERT(11)						

Opgave 103 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
	17	2		13		3				16

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 2k \text{ mod } 11$ og $h_2 = 1 + (4k \text{ mod } 10)$.

Angiv positionerne de fem elementer 1, 6, 7, 10 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 2, 3, 13, 16 og 17).

0	1	2	3	4	5	6	7	8	9	10
INSERT(1)										
	INSERT(6)									
		INSERT(7)								
			INSERT(10)							
				INSERT(11)						

Opgave 104 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
					15	18	10	7		19

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 4k \text{ mod } 11$ og $h_2 = 1 + (3k \text{ mod } 10)$.

Angiv positionerne de fem elementer 0, 1, 2, 3 og 8 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 7, 10, 15, 18 og 19).

0	1	2	3	4	5	6	7	8	9	10
INSERT(0)										
	INSERT(1)									
		INSERT(2)								
			INSERT(3)							
				INSERT(8)						

Opgave 105 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
11	0	18	5						15	

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 5k \text{ mod } 11$ og $h_2 = 1 + (3k \text{ mod } 10)$.

Angiv positionerne de fem elementer 3, 4, 6, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 5, 11, 15 og 18).

0	1	2	3	4	5	6	7	8	9	10
INSERT(3)										
	INSERT(4)									
		INSERT(6)								
			INSERT(9)							
				INSERT(10)						

Opgave 106 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
				1			21	13	10	19

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 4k \text{ mod } 11$ og $h_2 = 1 + (2k \text{ mod } 10)$.

Angiv positionerne de fem elementer 2, 3, 4, 5 og 8 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 1, 10, 13, 19 og 21).

0	1	2	3	4	5	6	7	8	9	10
INSERT(2)										
	INSERT(3)									
		INSERT(4)								
			INSERT(5)							
				INSERT(8)						

Opgave 107 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
13				2			9	15	20	

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 2k \text{ mod } 11$ og $h_2 = 1 + (2k \text{ mod } 10)$.

Angiv positionerne de fem elementer 0, 3, 4, 7 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 2, 9, 13, 15 og 20).

0 1 2 3 4 5 6 7 8 9 10

INSERT(0)

INSERT(3)

INSERT(4)

INSERT(7)

INSERT(10)

Opgave 108 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0				11	12		14			2

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 5k \text{ mod } 11$ og $h_2 = 1 + (3k \text{ mod } 10)$.

Angiv positionerne de fem elementer 3, 7, 8, 9 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 2, 11, 12 og 14).

0 1 2 3 4 5 6 7 8 9 10

INSERT(3)

INSERT(7)

INSERT(8)

INSERT(9)

INSERT(10)

Opgave 109 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
0	14			12		7			18	

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 4k \text{ mod } 11$ og $h_2 = 1 + (4k \text{ mod } 10)$.

Angiv positionerne de fem elementer 1, 3, 4, 8 og 10 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 0, 7, 12, 14 og 18).

0 1 2 3 4 5 6 7 8 9 10

INSERT(1)

INSERT(3)

INSERT(4)

INSERT(8)

INSERT(10)

Opgave 110 (Dobbelt hashing, 4 %)

0	1	2	3	4	5	6	7	8	9	10
	3		9	12	4			14		

I ovenstående hashtabel af størrelse 11 er anvendt *dobbelt hashing* med hashfunktionerne $h_1(k) = 4k \text{ mod } 11$ og $h_2 = 1 + (4k \text{ mod } 10)$.

Angiv positionerne de fem elementer 1, 2, 5, 8 og 11 vil blive indsat på i hashtabellen (for hver af indsættelserne antager vi at hashtabellen kun indeholder elementerne 3, 4, 9, 12 og 14).

0 1 2 3 4 5 6 7 8 9 10

INSERT(1)

INSERT(2)

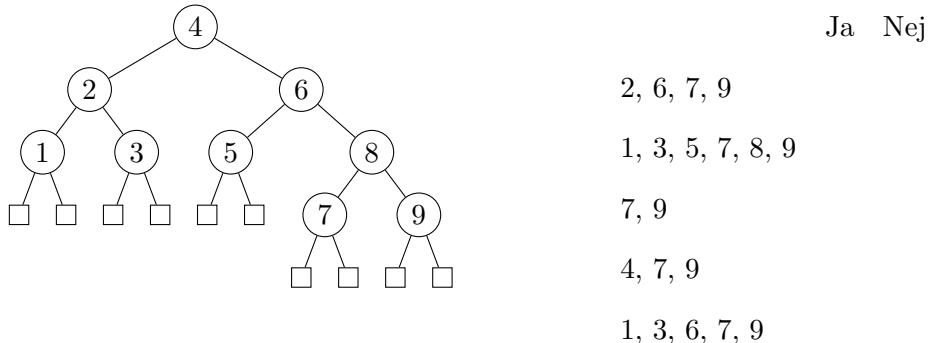
INSERT(5)

INSERT(8)

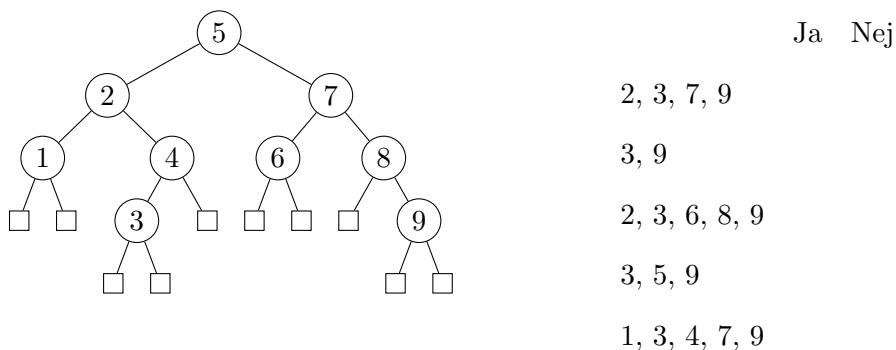
INSERT(11)

Opgave 111 (Rød-sort træ, 4 %)

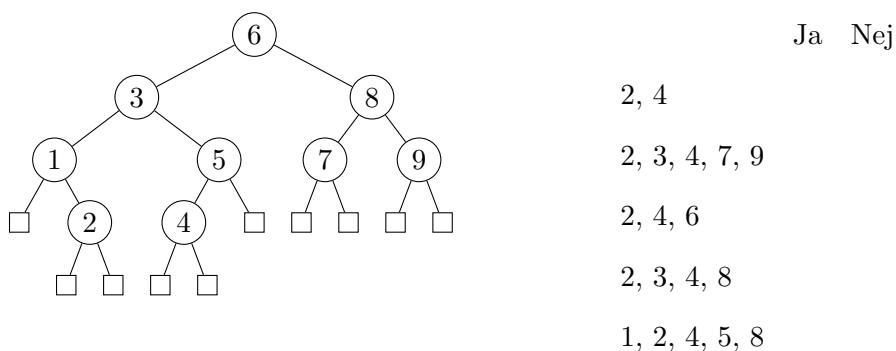
For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

**Opgave 112 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

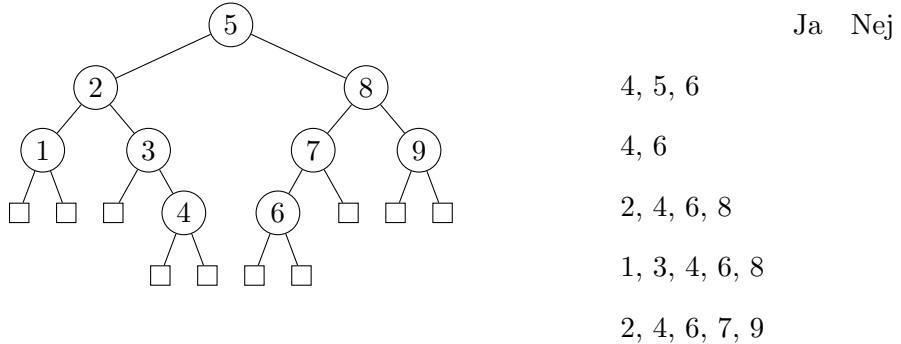
**Opgave 113 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

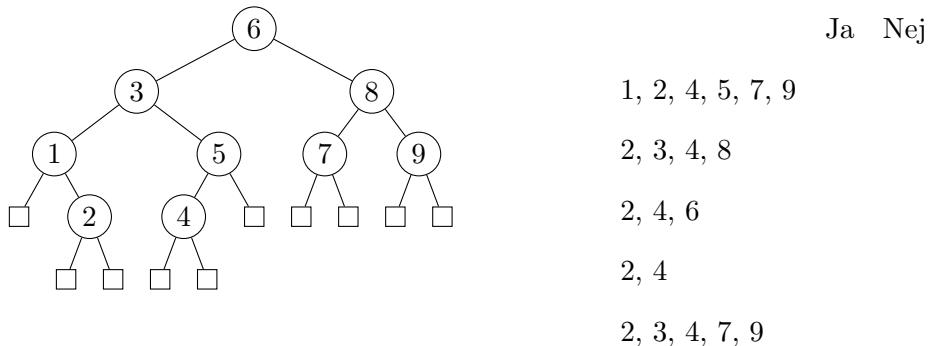


Opgave 114 (Rød-sort træ, 4 %)

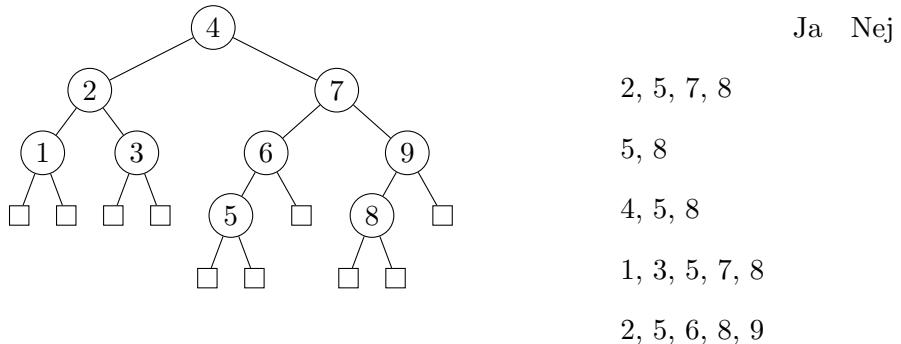
For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

**Opgave 115 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

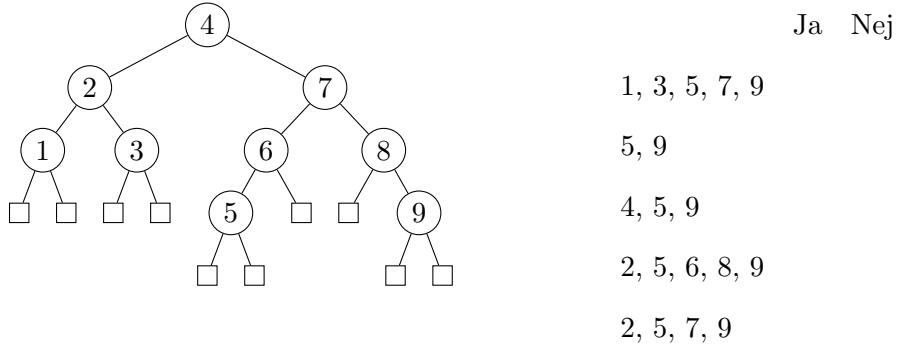
**Opgave 116 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

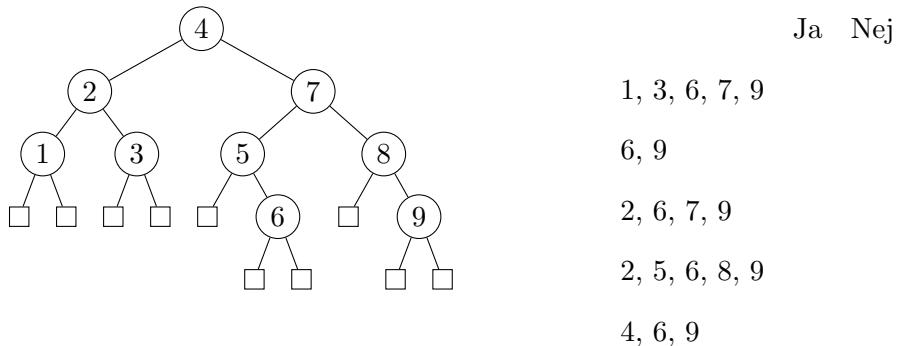


Opgave 117 (Rød-sort træ, 4 %)

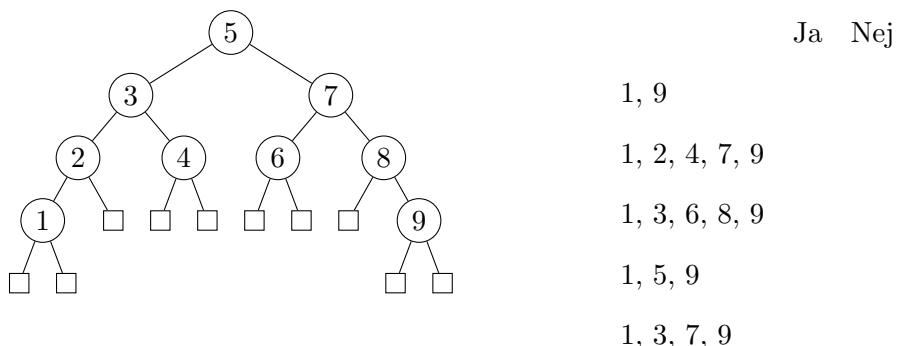
For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

**Opgave 118 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

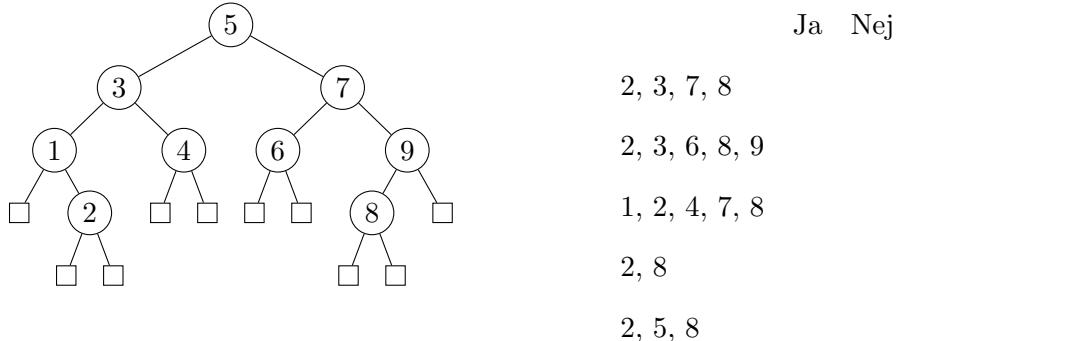
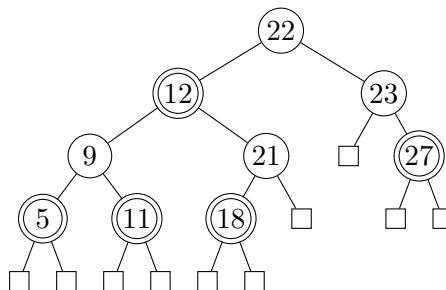
**Opgave 119 (Rød-sort træ, 4 %)**

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

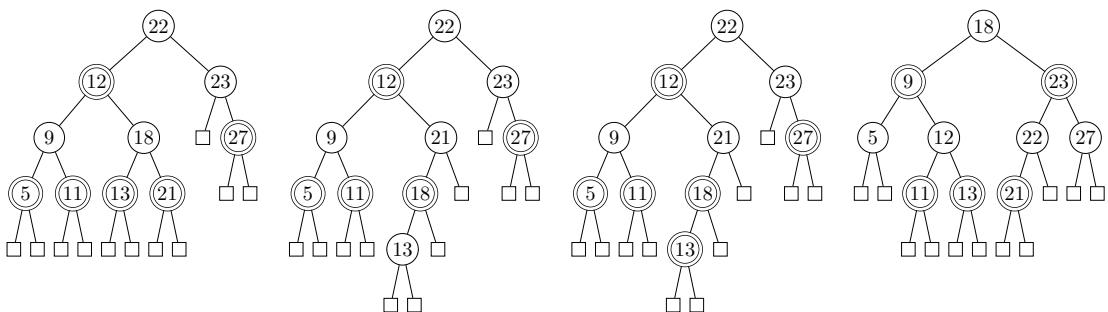


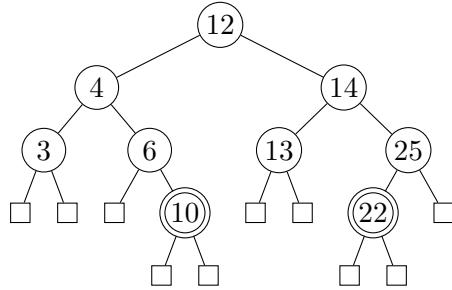
Opgave 120 (Rød-sort træ, 4 %)

For hver af nedenstående delmængder, angiv om nedenstående binære træ er et lovligt rød-sort træ hvis netop disse knuder farves røde.

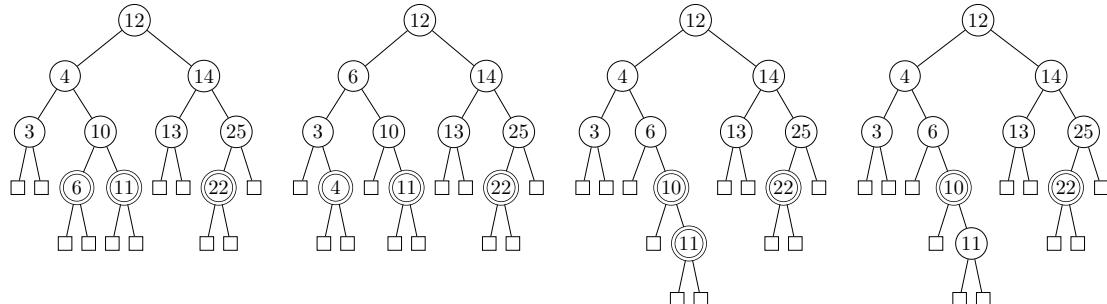
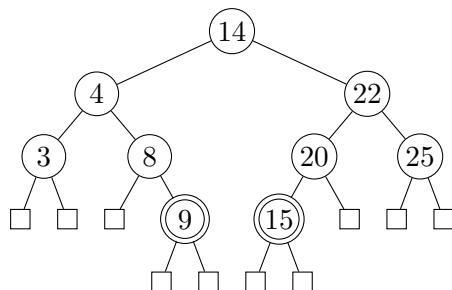
**Opgave 121 (Indsættelse i rød-sort træer, 4 %)**

Angiv det resulterende rød-sorte træ når man indsætter 13 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

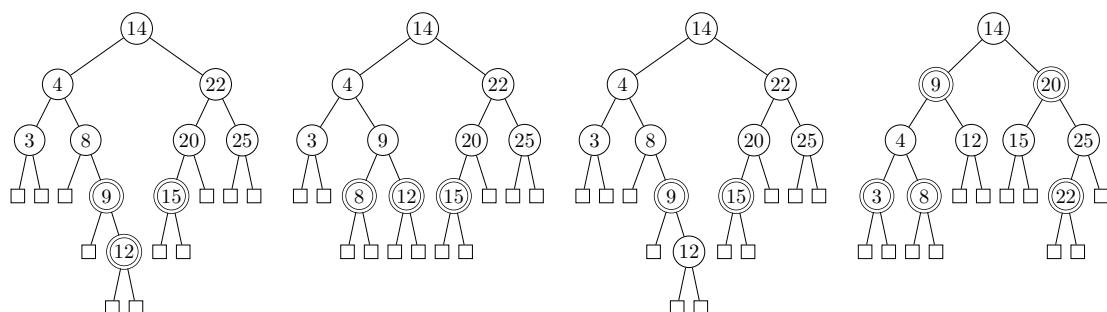


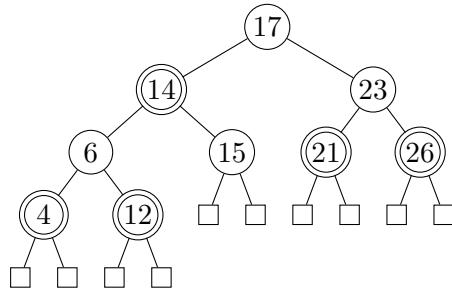
Opgave 122 (Indsættelse i rød-sort træer, 4 %)

Angiv det resulterende rød-sorte træ når man indsætter 11 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

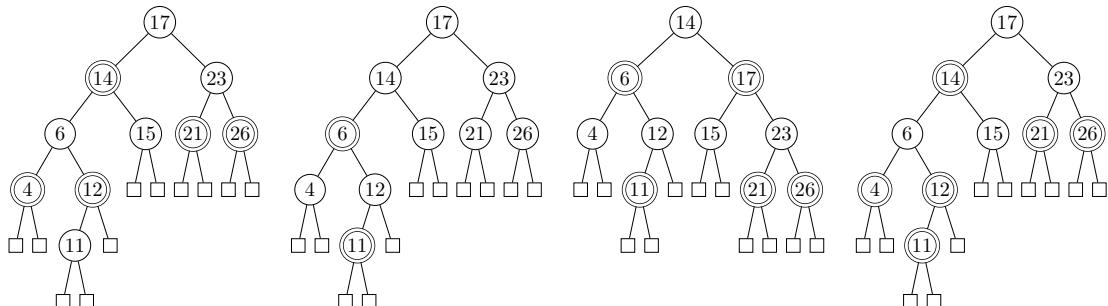
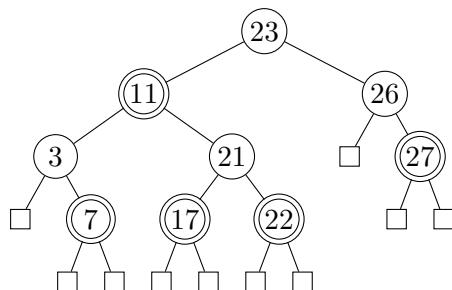
**Opgave 123 (Indsættelse i rød-sort træer, 4 %)**

Angiv det resulterende rød-sorte træ når man indsætter 12 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

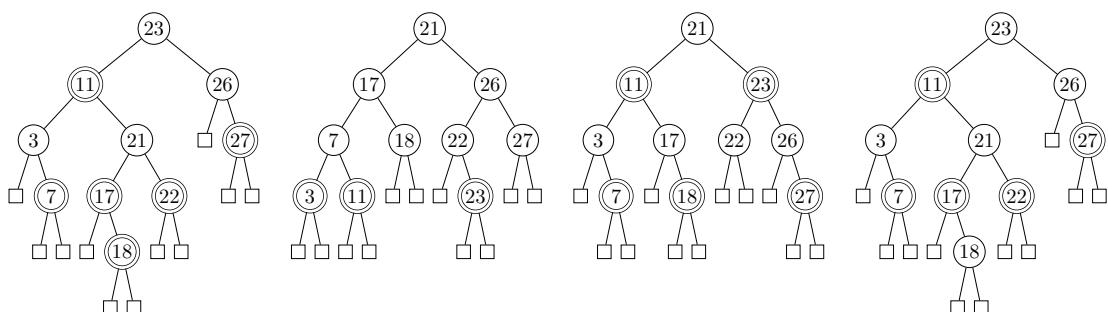


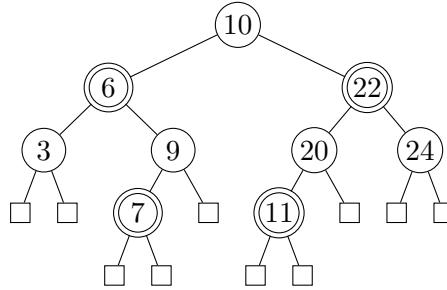
Opgave 124 (Indsættelse i rød-sort træer, 4 %)

Angiv det resulterende rød-sorte træ når man indsætter 11 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

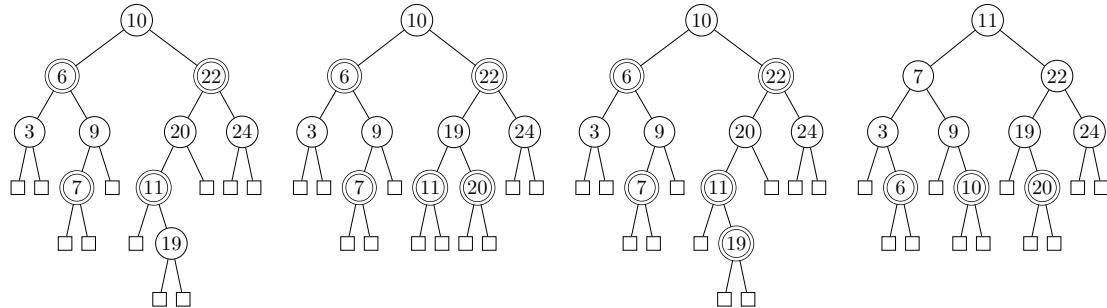
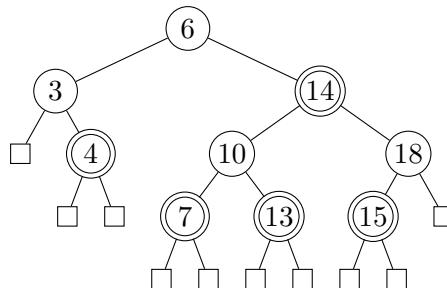
**Opgave 125 (Indsættelse i rød-sort træer, 4 %)**

Angiv det resulterende rød-sorte træ når man indsætter 18 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

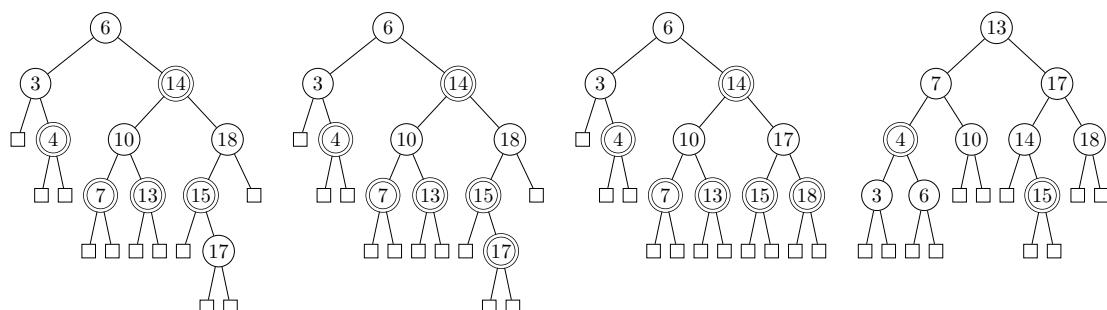


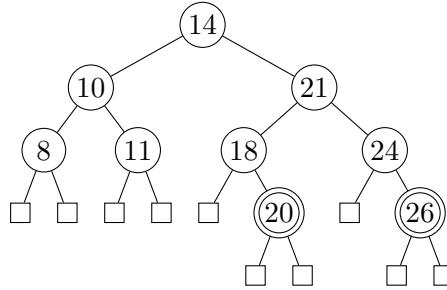
Opgave 126 (Indsættelse i rød-sort træer, 4 %)

Angiv det resulterende rød-sorte træ når man indsætter 19 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

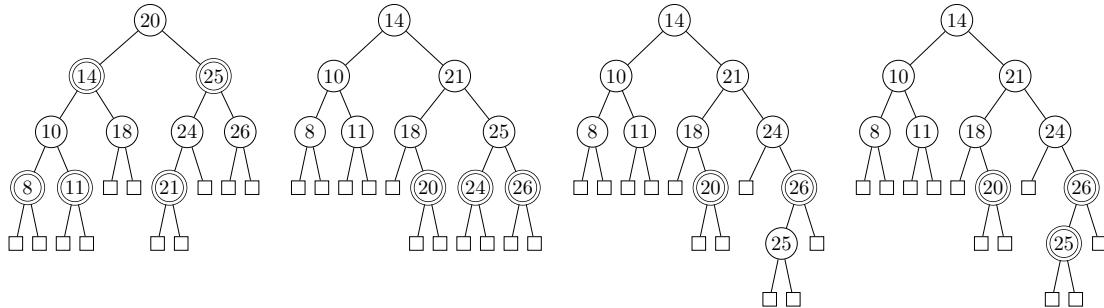
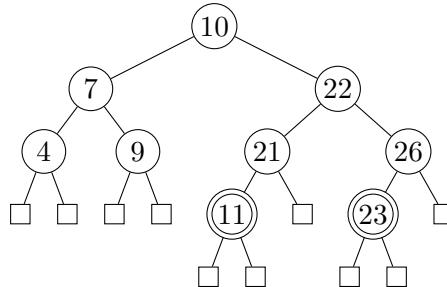
**Opgave 127 (Indsættelse i rød-sorte træer, 4 %)**

Angiv det resulterende rød-sorte træ når man indsætter 17 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

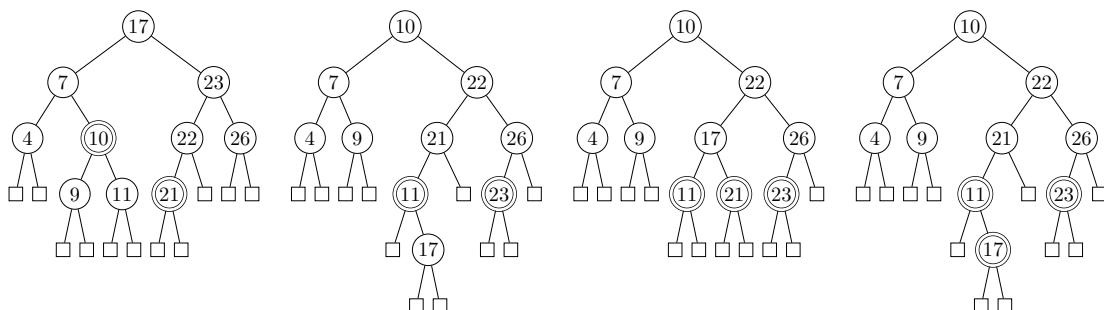


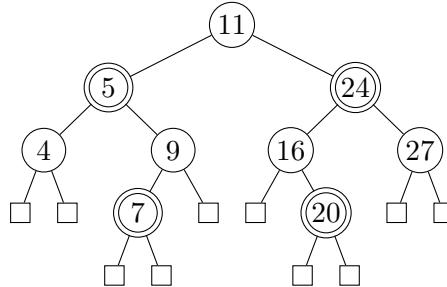
Opgave 128 (Indsættelse i rød-sort træer, 4 %)

Angiv det resulterende rød-sorte træ når man indsætter 25 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

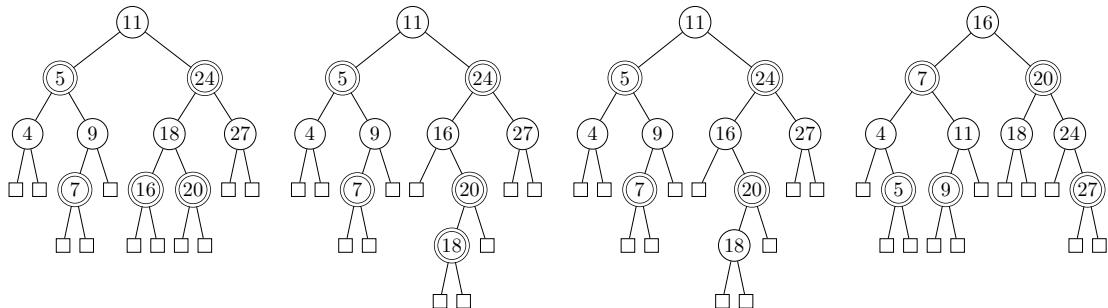
**Opgave 129 (Indsættelse i rød-sort træer, 4 %)**

Angiv det resulterende rød-sorte træ når man indsætter 17 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).



Opgave 130 (Indsættelse i rød-sort træer, 4 %)

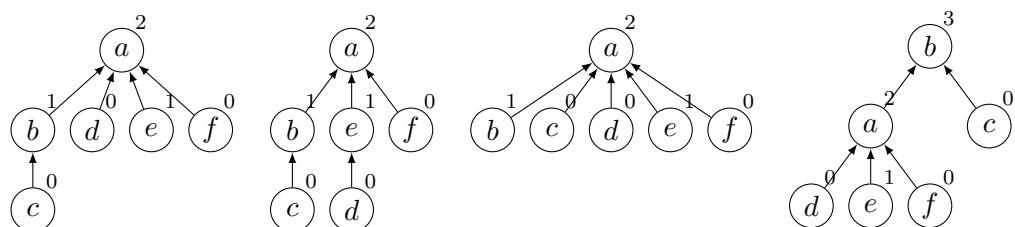
Angiv det resulterende rød-sorte træ når man indsætter 18 i ovenstående rød-sorte træ (dobbeltcirkler angiver røde knuder).

**Opgave 131 (Union-find, 4 %)**

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET(a)
MAKESET(b)
MAKESET(c)
MAKESET(d)
MAKESET(e)
MAKESET(f)
UNION(d, e)
UNION(f, a)
UNION(d, f)
UNION(c, b)
UNION(d, c)
FIND-SET(b)
  
```



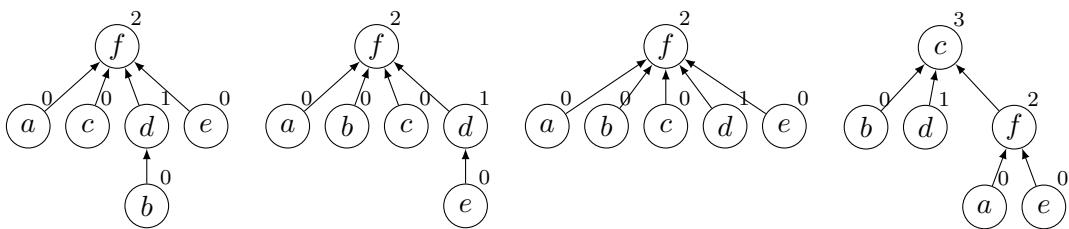
Opgave 132 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $e, d$ )
UNION( $b, e$ )
UNION( $a, f$ )
UNION( $e, f$ )
UNION( $e, c$ )
FIND-SET( $b$ )

```

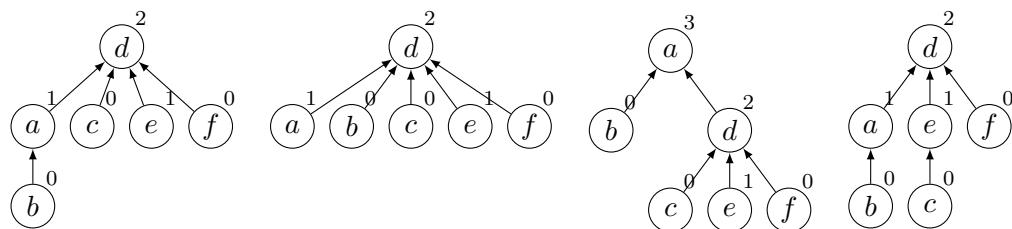
**Opgave 133 (Union-find, 4 %)**

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $c, e$ )
UNION( $f, d$ )
UNION( $c, f$ )
UNION( $b, a$ )
UNION( $c, b$ )
FIND-SET( $a$ )

```



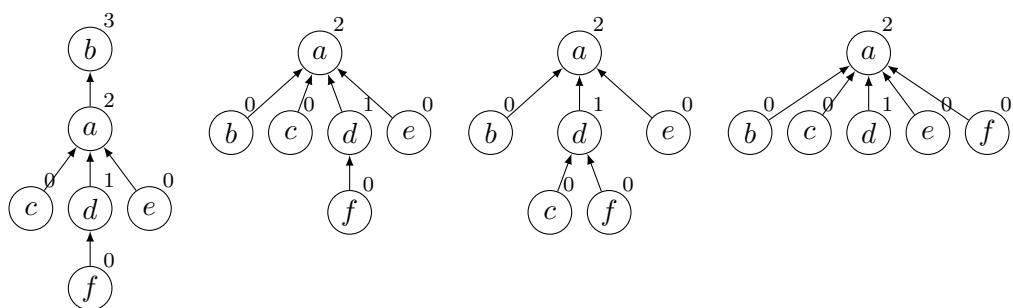
Opgave 134 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $c, d$ )
UNION( $f, d$ )
UNION( $e, a$ )
UNION( $f, e$ )
UNION( $c, b$ )
FIND-SET( $b$ )

```



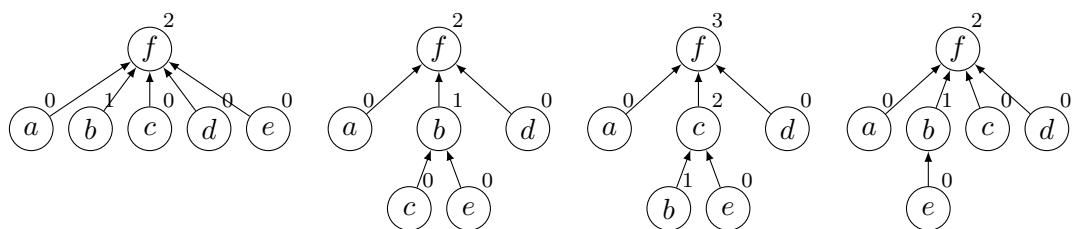
Opgave 135 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $e, b$ )
UNION( $e, c$ )
UNION( $a, f$ )
UNION( $e, a$ )
UNION( $d, c$ )
FIND-SET( $a$ )

```



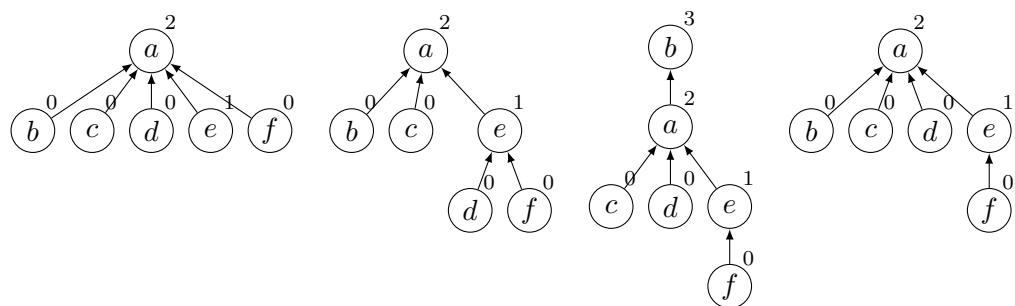
Opgave 136 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $d, e$ )
UNION( $f, d$ )
UNION( $c, a$ )
UNION( $f, a$ )
UNION( $d, b$ )
FIND-SET( $b$ )

```



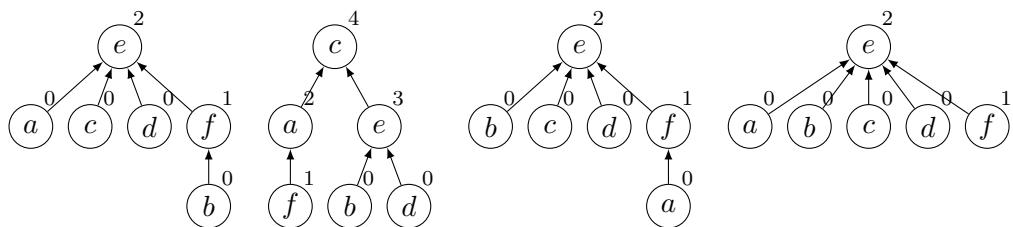
Opgave 137 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESSET( $a$ )
MAKESSET( $b$ )
MAKESSET( $c$ )
MAKESSET( $d$ )
MAKESSET( $e$ )
MAKESSET( $f$ )
UNION( $b, f$ )
UNION( $b, a$ )
UNION( $d, e$ )
UNION( $b, d$ )
UNION( $b, c$ )
FIND-SET( $a$ )

```

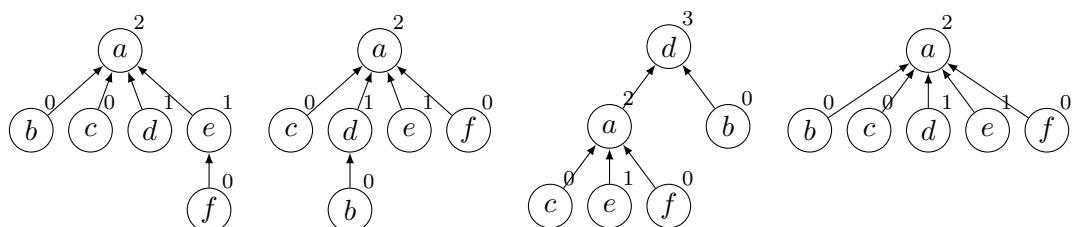
**Opgave 138 (Union-find, 4 %)**

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESSET( $a$ )
MAKESSET( $b$ )
MAKESSET( $c$ )
MAKESSET( $d$ )
MAKESSET( $e$ )
MAKESSET( $f$ )
UNION( $f, e$ )
UNION( $c, a$ )
UNION( $e, c$ )
UNION( $b, d$ )
UNION( $f, d$ )
FIND-SET( $b$ )

```



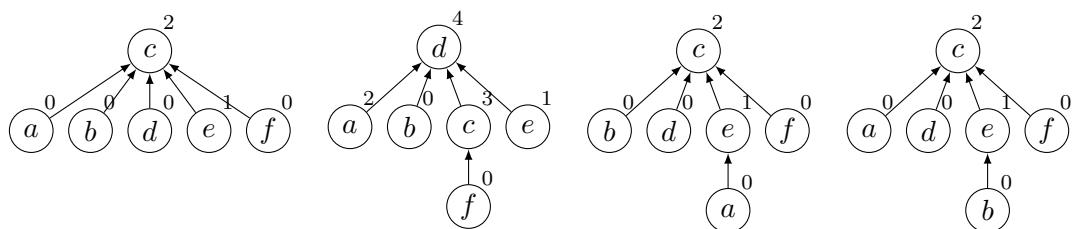
Opgave 139 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $b, e$ )
UNION( $e, a$ )
UNION( $f, c$ )
UNION( $e, f$ )
UNION( $a, d$ )
FIND-SET( $b$ )

```



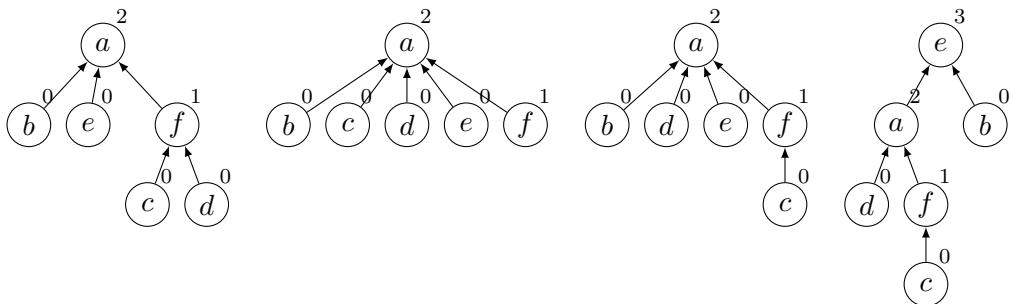
Opgave 140 (Union-find, 4 %)

Angiv den resulterende union-find struktur efter nedenstående sekvens af operationer, når der anvendes union-by-rank og stikomprimering.

```

MAKESET( $a$ )
MAKESET( $b$ )
MAKESET( $c$ )
MAKESET( $d$ )
MAKESET( $e$ )
MAKESET( $f$ )
UNION( $c, f$ )
UNION( $d, c$ )
UNION( $b, a$ )
UNION( $c, a$ )
UNION( $d, e$ )
FIND-SET( $b$ )

```

**Opgave 141 (Rekursionsligninger, 4 %)**

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = T(n - 1) + 2$$

$$T(n) = 4 \cdot T(n/4) + n$$

$$T(n) = 2 \cdot T(n/5) + n$$

$$T(n) = 4 \cdot T(n/2) + n^2$$

$$T(n) = 2 \cdot T(n/3) + n^3$$

Opgave 142 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = T(n - 1) + n$$

$$T(n) = 2 \cdot T(n/4) + n^2$$

$$T(n) = 3 \cdot T(n/9) + 1$$

$$T(n) = T(n - 1) + 3$$

$$T(n) = 9 \cdot T(n/3) + n^2$$

Opgave 143 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 3 \cdot T(n/9) + 2$$

$$T(n) = T(n - 1) + \log n$$

$$T(n) = 4 \cdot T(n/2) + 1$$

$$T(n) = T(n - 1) + n^2$$

$$T(n) = 4 \cdot T(n/2) + n^2$$

Opgave 144 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 3 \cdot T(n/9) + 1$$

$$T(n) = 2 \cdot T(n/3) + n$$

$$T(n) = 4 \cdot T(n/2) + 1$$

$$T(n) = 4 \cdot T(n/2) + n^2$$

$$T(n) = T(n - 1) + n^2$$

Opgave 145 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 4 \cdot T(n/2) + n^2$$

$$T(n) = T(n/5) + 5$$

$$T(n) = 3 \cdot T(n/4) + n^3$$

$$T(n) = 3 \cdot T(n/4) + n$$

$$T(n) = 2 \cdot T(n/4) + 1$$

Opgave 146 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 4 \cdot T(n/4) + n$$

$$T(n) = 4 \cdot T(n/2) + n^2$$

$$T(n) = T(n/3) + 2$$

$$T(n) = 4 \cdot T(n/5) + n^3$$

$$T(n) = 9 \cdot T(n/3) + 3$$

Opgave 147 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 2 \cdot T(n/3) + n^2$$

$$T(n) = 3 \cdot T(n/3) + n$$

$$T(n) = 2 \cdot T(n/4) + 3$$

$$T(n) = 9 \cdot T(n/3) + n^2$$

$$T(n) = 4 \cdot T(n/2) + 1$$

Opgave 148 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 9 \cdot T(n/3) + n^2$$

$$T(n) = 5 \cdot T(n/5) + n$$

$$T(n) = 9 \cdot T(n/3) + 1$$

$$T(n) = T(n/2) + 3$$

$$T(n) = 3 \cdot T(n/9) + 1$$

Opgave 149 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

$$T(n) = 2 \cdot T(n/4) + n$$

$$T(n) = 2 \cdot T(n/4) + 1$$

$$T(n) = 4 \cdot T(n/2) + 2$$

$$T(n) = 9 \cdot T(n/3) + 2$$

$$T(n) = 8 \cdot T(n/2) + 3$$

Opgave 150 (Rekursionsligninger, 4 %)

Angiv løsningen for hver af nedenstående rekursionsligninger, hvor $T(n) = 1$ for $n \leq 1$.

$$\Theta(\log n) \quad \Theta(\sqrt{n}) \quad \Theta(n) \quad \Theta(n \log n) \quad \Theta(n^2) \quad \Theta(n^2 \log n) \quad \Theta(n^3)$$

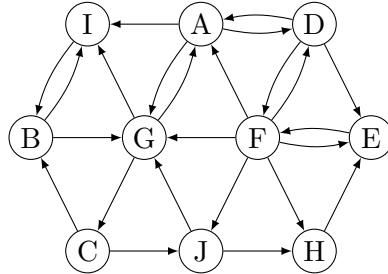
$$T(n) = 4 \cdot T(n/2) + n^2$$

$$T(n) = 2 \cdot T(n/4) + 3$$

$$T(n) = T(n - 1) + n^2$$

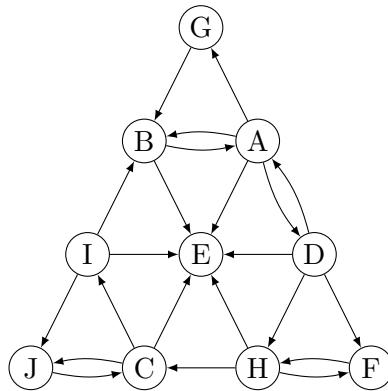
$$T(n) = T(n - 1) + \log n$$

$$T(n) = 2 \cdot T(n/4) + n^3$$

Opgave 151 (BFS, 4 %)

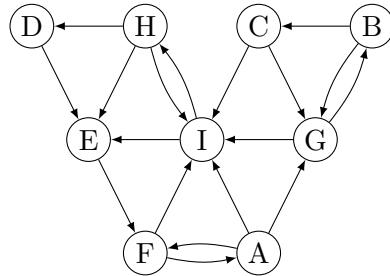
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

ADIGEFBCJH ADGIEFCBHJ ADEFGCBIJH ADGIEFCBJH

Opgave 152 (BFS, 4 %)

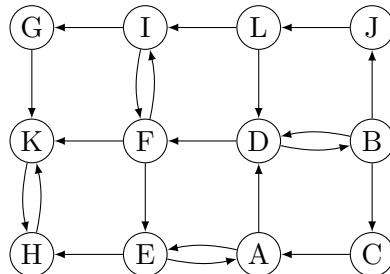
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

AGEDBHFCJI ABDEGFHCJI ABEDFHCIJG ABDEGFHCIJ

Opgave 153 (BFS, 4 %)

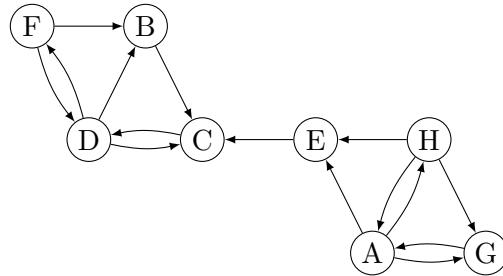
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A F G I B H E C D A I F G E H B D C A F G I B E H C D A F I E H D G B C

Opgave 154 (BFS, 4 %)

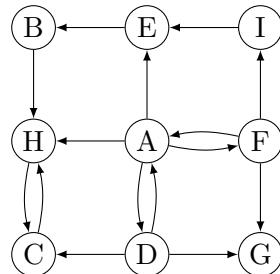
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A D B C J L I F E H K G A E D H B F K J C I L G A D E B F H C J K I L G A D E B F H C J I K L G

Opgave 155 (BFS, 4 %)

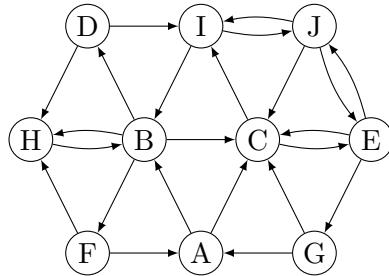
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver udtaget af køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A E C D B F G H A E G H C D F B A E G H C D B F A G H E C D B F

Opgave 156 (BFS, 4 %)

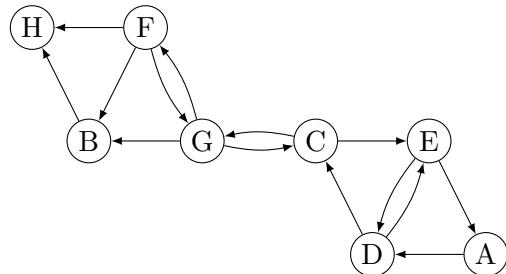
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver udtaget af køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A F H E D I G C B A D E F H C G B I A D E F H G C B I A D C H G E B F I

Opgave 157 (BFS, 4 %)

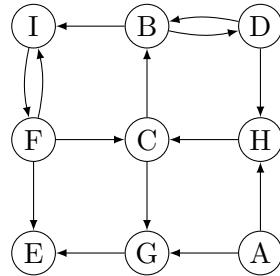
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A B C D F H I E J G A B C E G J I D H F A C B I E D H F J G A B C D F H E I G J

Opgave 158 (BFS, 4 %)

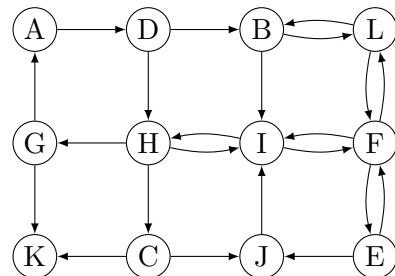
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A D C E G F B H A D C E G B F H A D C E G B H F A D E C G B F H

Opgave 159 (BFS, 4 %)

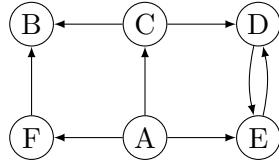
For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver udtaget af køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A G H E C B D I F A G H E C B I D F A G E H C B D I F A H G C E B I D F

Opgave 160 (BFS, 4 %)

For et bredde først gennemløb (BFS) af ovenstående graf **startende i knuden A**, angiv rækkefølgen knuderne bliver indsat i køen Q i BFS-algoritmen. Det antages, at grafen er givet ved incidenslister, hvor incidenslisterne er sorteret alfabetisk.

A D H B C G I L J K F E A D B H I L C G F J K E A D B H I L C G F K J E A D B I F E J L H C K G

Opgave 161 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

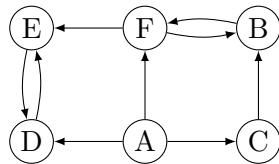
(A,C) (A,E) (A,F) (C,B) (E,D)

(A,C) (A,F) (C,B) (C,D) (D,E)

(A,C) (A,E) (A,F) (E,D) (F,B)

(A,C) (A,E) (A,F) (C,B) (C,D)

(A,C) (A,E) (A,F) (C,D) (F,B)

Opgave 162 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

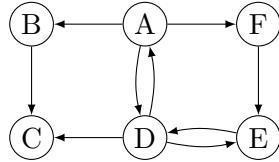
(A,C) (A,D) (A,F) (F,B) (F,E)

(A,C) (A,D) (A,F) (C,B) (D,E)

(A,C) (A,D) (A,F) (D,E) (F,B)

(A,C) (A,D) (A,F) (C,B) (F,E)

(A,C) (A,D) (B,F) (C,B) (D,E)

Opgave 163 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

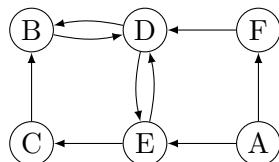
(A,B) (A,D) (A,F) (B,C) (D,E)

(A,B) (A,F) (B,C) (E,D) (F,E)

(A,B) (A,D) (A,F) (D,C) (F,E)

(A,B) (A,D) (A,F) (B,C) (F,E)

(A,B) (A,F) (D,C) (E,D) (F,E)

Opgave 164 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

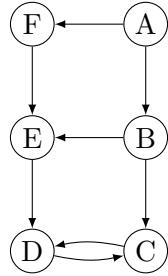
(A,E) (A,F) (C,B) (E,C) (E,D)

(A,E) (A,F) (D,B) (E,C) (E,D)

(A,E) (A,F) (B,D) (C,B) (E,C)

(A,F) (D,B) (D,E) (E,C) (F,D)

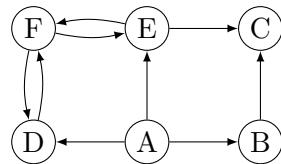
(A,E) (A,F) (D,B) (E,C) (F,D)

Opgave 165 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

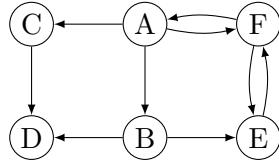
- (A,B) (A,F) (B,C) (B,E) (C,D)
- (A,B) (A,F) (B,C) (B,E) (E,D)
- (A,B) (A,F) (B,C) (E,D) (F,E)
- (A,B) (A,F) (B,E) (D,C) (E,D)
- (A,B) (A,F) (D,C) (E,D) (F,E)

Opgave 166 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

- (A,B) (A,D) (A,E) (B,C) (D,F)
- (A,B) (A,D) (A,E) (D,F) (E,C)
- (A,B) (A,E) (E,C) (E,F) (F,D)
- (A,B) (A,D) (B,C) (D,F) (F,E)
- (A,B) (A,D) (A,E) (E,C) (E,F)

Opgave 167 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

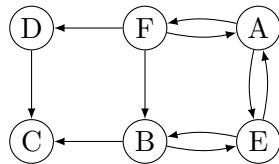
(A,B) (A,C) (A,F) (B,D) (F,E)

(A,B) (A,C) (A,F) (C,D) (F,E)

(A,B) (A,C) (A,F) (B,D) (B,E)

(A,B) (A,C) (B,D) (B,E) (E,F)

(A,B) (A,C) (A,F) (B,E) (C,D)

Opgave 168 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

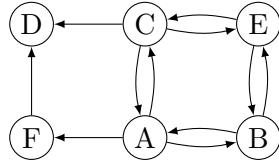
(A,E) (A,F) (B,C) (F,B) (F,D)

(A,F) (B,C) (B,E) (F,B) (F,D)

(A,E) (A,F) (D,C) (F,B) (F,D)

(A,E) (A,F) (B,C) (E,B) (F,D)

(A,E) (A,F) (D,C) (E,B) (F,D)

Opgave 169 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

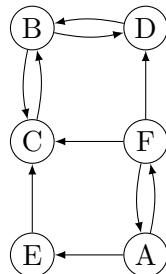
(A,B) (A,F) (B,E) (C,D) (E,C)

(A,B) (A,C) (A,F) (C,E) (F,D)

(A,B) (A,C) (A,F) (C,D) (C,E)

(A,C) (A,F) (C,E) (E,B) (F,D)

(A,C) (A,F) (C,D) (C,E) (E,B)

Opgave 170 (Lovlige bredde først træer, 4 %)

Angiv for hver af nedenstående mængder af kanter om de udgør et lovligt BFS træ for et bredde først gennemløb af ovenstående graf **startende i knuden A** og for en vilkårlig ordning af grafens incidenslister.

Ja Nej

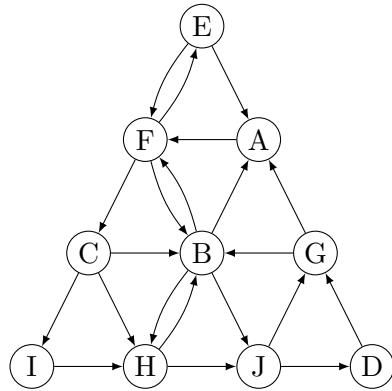
(A,E) (A,F) (C,B) (E,C) (F,D)

(A,E) (A,F) (C,B) (F,C) (F,D)

(A,E) (A,F) (D,B) (F,C) (F,D)

(A,E) (A,F) (B,C) (D,B) (F,D)

(A,E) (A,F) (D,B) (E,C) (F,D)

Opgave 171 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **finishing time**.

EGDJBHICFA GDJHBICEFA GDIJHECBFA EICGDJHBFA

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

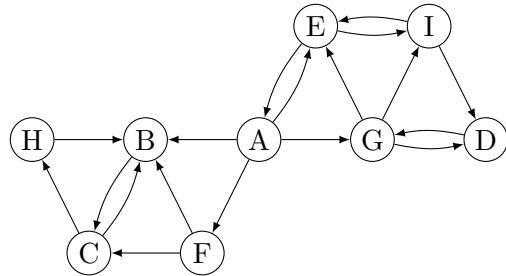
Tree edge Back edge Cross edge Forward edge

(H, B)

(D, G)

(B, J)

(I, H)

Opgave 172 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A F C H B G E I D A B C H E I D G F A B E F G C I D H A F E I D G B C H

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

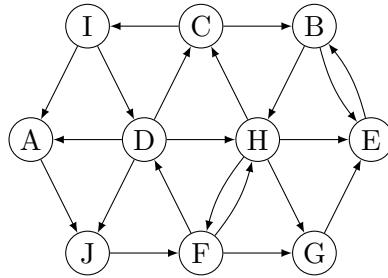
Tree edge Back edge Cross edge Forward edge

(G, D)

(F, B)

(A, G)

(A, F)

Opgave 173 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **finishing time**.

IBECHGDFJA IGHEBCDFJA EGHBICDFJA IEBCGHDFJA

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

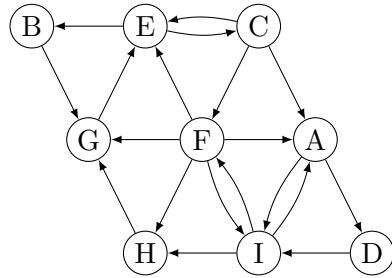
Tree edge Back edge Cross edge Forward edge

(H, G)

(D, H)

(G, E)

(E, B)

Opgave 174 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A D I F H E C B G A D I H G E B C F A D I F H E G B C A D I F E B G C H

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

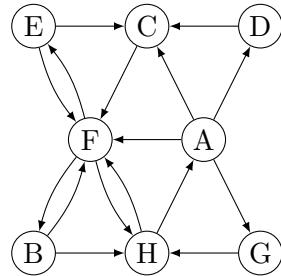
Tree edge Back edge Cross edge Forward edge

(G, E)

(H, G)

(F, H)

(A, I)

Opgave 175 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A G D C F E B H A C F B H E D G A C D F G B E H A C F H E B D G

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

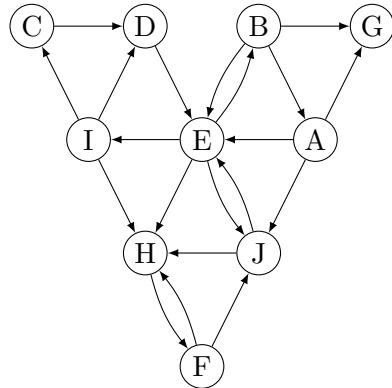
Tree edge Back edge Cross edge Forward edge

(D, C)

(F, H)

(B, H)

(H, A)

Opgave 176 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A J H F E I C D B G A E G J B H I F C D A E I C D H F J B G A E B G H F J I C D

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

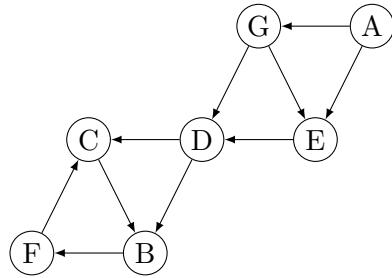
Tree edge Back edge Cross edge Forward edge

(I, D)

(I, H)

(B, E)

(E, I)

Opgave 177 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **finishing time**.

F C B D G E A F B C D E G A G C F B D E A C F B D E G A

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

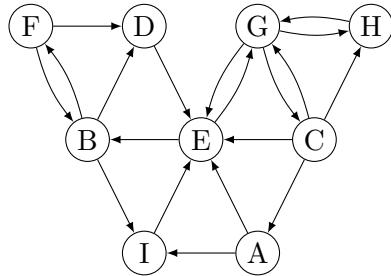
Tree edge Back edge Cross edge Forward edge

(F, C)

(G, E)

(C, B)

(D, C)

Opgave 178 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i knuden **A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A E I B G D F C H A I E B D F G H C A E G C H B I F D A E B D F I G C H

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

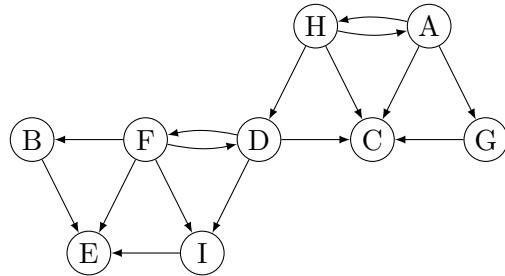
Tree edge Back edge Cross edge Forward edge

(F, D)

(H, G)

(B, F)

(A, I)

Opgave 179 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A C H D I E F B G A H D F I B E G C A C G H D F B E I A C G H D F I B E

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

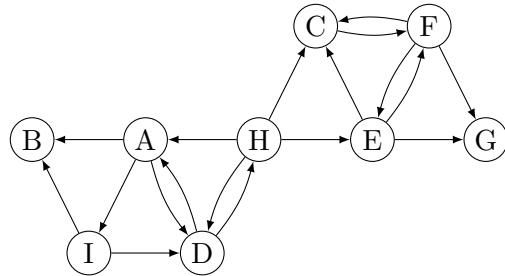
Tree edge Back edge Cross edge Forward edge

(D, F)

(F, E)

(H, C)

(F, D)

Opgave 180 (DFS, 4 %)

Betræt et dybde først gennemløb (DFS) af ovenstående graf, hvor DFS-gennemløbet starter i **knuden A**, hvor de udgående kanter til en knude besøges i alfabetisk rækkefølge. Angiv i hvilken rækkefølge knuderne får tildelt **discovery time**.

A B D I H C E F G A B D H C F G E I A I D H C F E G B A B D H C F E G I

Angiv for hver af nedenstående kanter hvilken type kanten bliver i DFS gennemløbet.

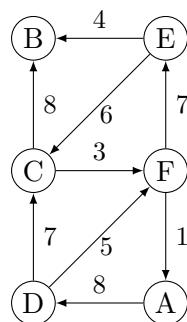
Tree edge Back edge Cross edge Forward edge

(H, A)

(F, E)

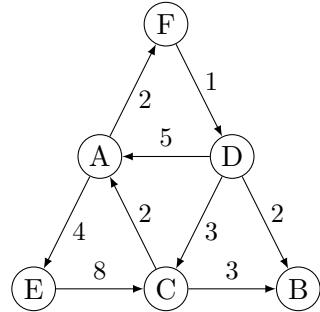
(I, B)

(F, G)

Opgave 181 (Dijkstras algoritme, 4 %)

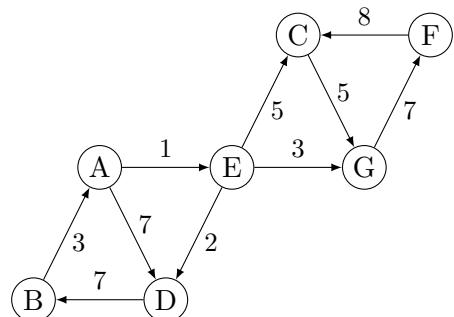
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A D C B F E A D F E B C A D F C E B A D C F B E

Opgave 182 (Dijkstras algoritme, 4 %)

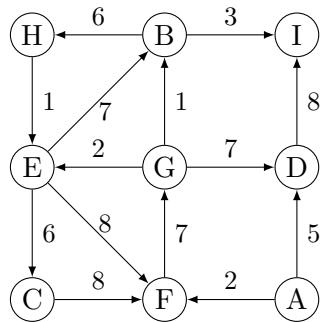
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A E F C D B A F D E B C A E C B F D A F D B C E

Opgave 183 (Dijkstras algoritme, 4 %)

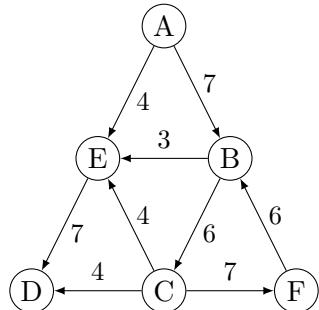
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A E D G C F B A D B E C G F A D E B C G F A E D G C B F

Opgave 184 (Dijkstras algoritme, 4 %)

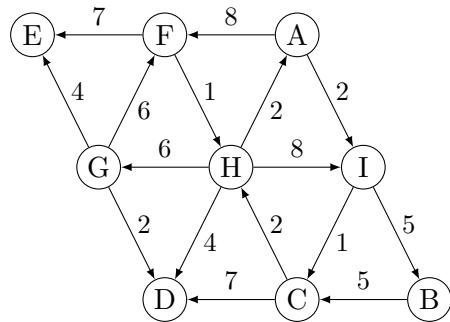
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A D I F G B H E C A F D G B I H E C A D F I G B E H C A F D G B E I H C

Opgave 185 (Dijkstras algoritme, 4 %)

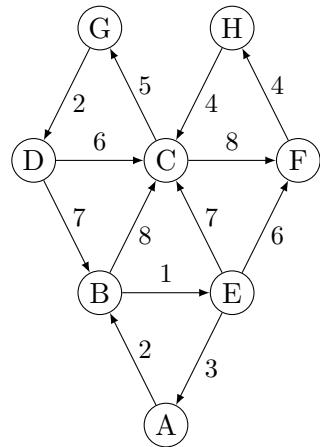
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A E B D C F A B E C D F A B C D E F A E B C D F

Opgave 186 (Dijkstras algoritme, 4 %)

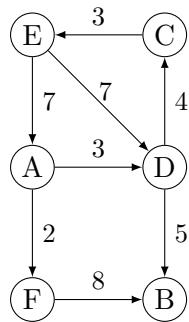
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knode A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A F E H D G I B C A I C H B F E D G A I C H B F D G E A F I E H B C D G

Opgave 187 (Dijkstras algoritme, 4 %)

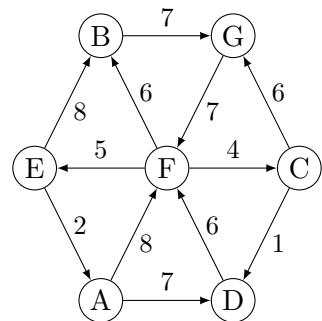
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knode A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

A B C F H G D E A B E F C H G D A B E F H C G D A B C E F G H D

Opgave 188 (Dijkstras algoritme, 4 %)

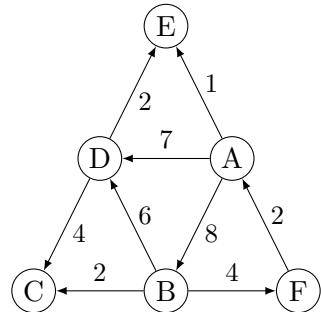
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

AFDCBE ADFBCE AFDCEB ADBCEF

Opgave 189 (Dijkstras algoritme, 4 %)

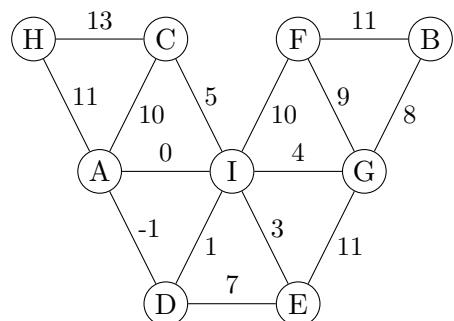
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

ADFCEBG ADFBCEG ADFCGEB ADFBGCE

Opgave 190 (Dijkstras algoritme, 4 %)

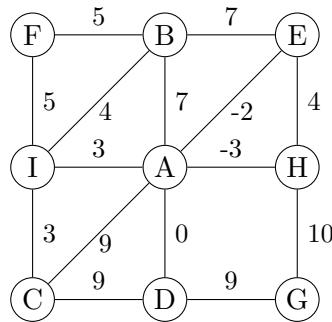
Antag Dijkstras algoritme anvendes til at finde korteste afstande fra **knuden A** til alle knuder i ovenstående graf. Angiv hvilken rækkefølge knuderne bliver taget ud af prioritetskøen i Dijkstra's algoritme.

ABDECFC ABCDEF AEDBCF AEDCBF

Opgave 191 (Prims algoritme, 4 %)

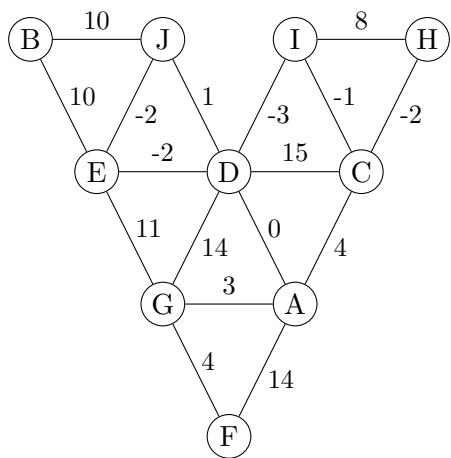
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **knuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

ADIEGBFCH ADIEGCBFH ADIEGCHBF ADIEGCFHB

Opgave 192 (Prims algoritme, 4 %)

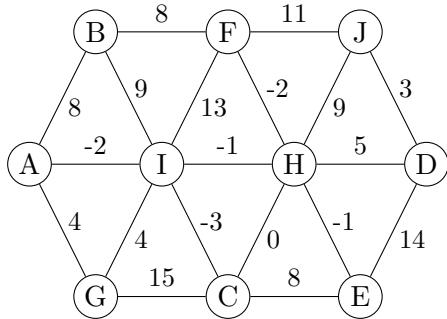
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A H E D I C B F G A H E D I C B G F A H E B I C D G F A H E D I B C G F

Opgave 193 (Prims algoritme, 4 %)

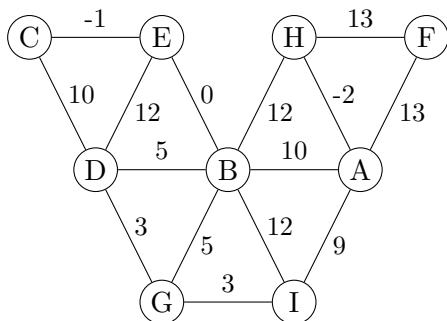
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A D I C H E J B G F A D I C H E J G B F A D I E J C H G F B A D I C H E J G F B

Opgave 194 (Prims algoritme, 4 %)

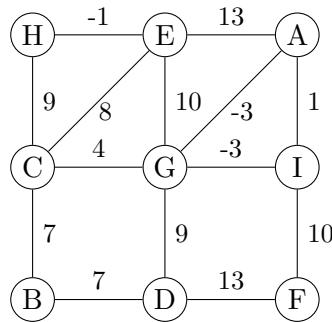
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

AICHFEDGJB AICHFBJD EG AICHFEGDJB AICHFEDBGJ

Opgave 195 (Prims algoritme, 4 %)

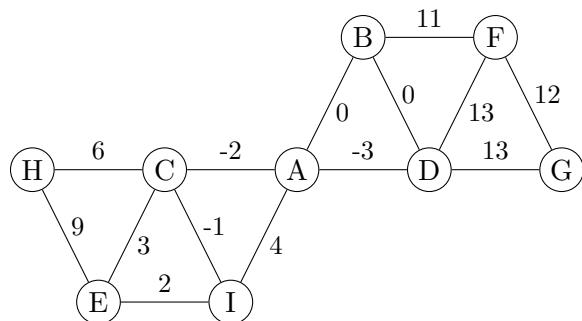
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

AHIGDBECF AHBECDGIF AHIBECFGD AHIGFD BEC

Opgave 196 (Prims algoritme, 4 %)

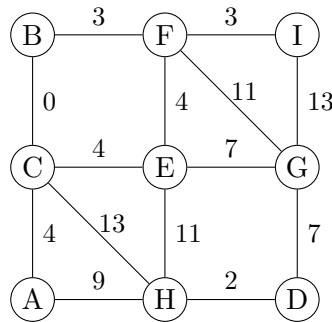
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A G I F D B C E H A G I C F B E H D A G I C F D E H B A G I C B D E H F

Opgave 197 (Prims algoritme, 4 %)

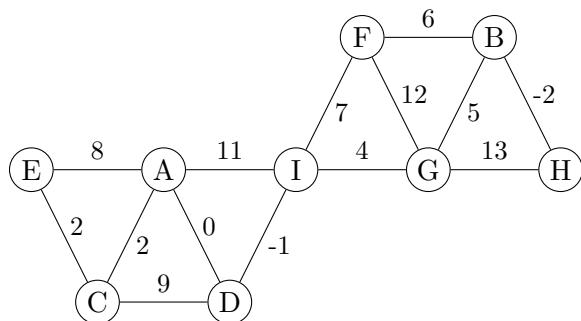
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A D B F G C I E H A D C I E B H F G A D C I B E H F G A D B C I E H F G

Opgave 198 (Prims algoritme, 4 %)

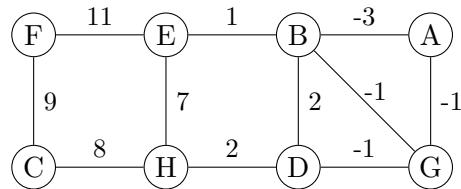
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A C B F I E G D H A C B F E H I D G A C B F E I G D H A C B F I G D H E

Opgave 199 (Prims algoritme, 4 %)

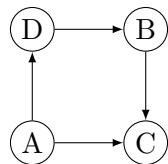
Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A D I G B H F C E A D I C E G B H F A D I C G E F B H A D I C G E B H F

Opgave 200 (Prims algoritme, 4 %)

Antag Prims algoritme anvendes til at finde et minimum udspændende træ for ovenstående graf, og algoritmen starter i **k nuden A**. Angiv hvilken rækkefølge knuderne bliver inkludert i det minimum udspændende træ (taget ud af prioritetskøen i Prims algoritme).

A B G D H E C F A B E G D H C F A B G D E H C F A B G D H E F C

Opgave 201 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

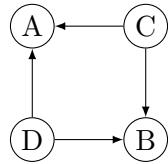
C D B A

A D B C

A D C B

A B D C

B D A C

Opgave 202 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

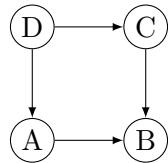
C D A B

D B A C

A C B D

C D B A

D C B A

Opgave 203 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

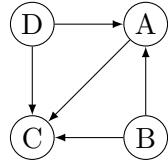
D B C A

D A C B

D B A C

D C A B

B C A D

Opgave 204 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

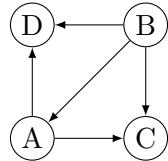
A B D C

D C A B

D B A C

A D B C

B D A C

Opgave 205 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

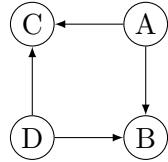
A B C D

B A D C

B A C D

B C D A

B D C A

Opgave 206 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

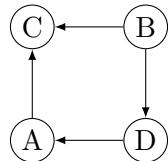
D A C B

D C A B

A D C B

D A B C

D B A C

Opgave 207 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

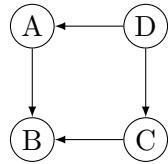
B D A C

C D A B

A D B C

B A D C

B C A D

Opgave 208 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

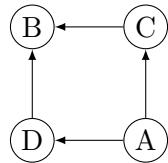
D A C B

D A B C

B A C D

B C A D

D C A B

Opgave 209 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

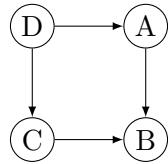
B C D A

D C A B

A C D B

A D C B

B D C A

Opgave 210 (Topologisk sortering, 4 %)

Angiv for hver af nedenstående ordninger af knuderne i ovenstående graf om det er en lovlig topologisk sortering.

Ja Nej

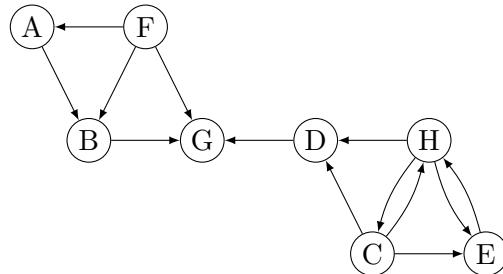
D C A B

C D A B

D A B C

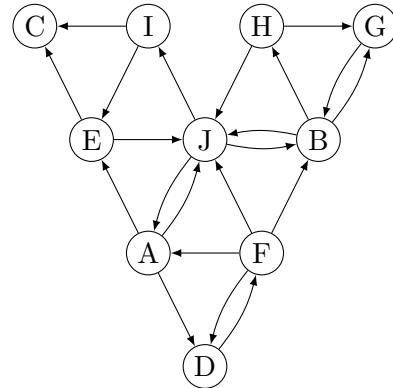
D B C A

D A C B

Opgave 211 (Stærke sammenhængskomponenter, 4 %)

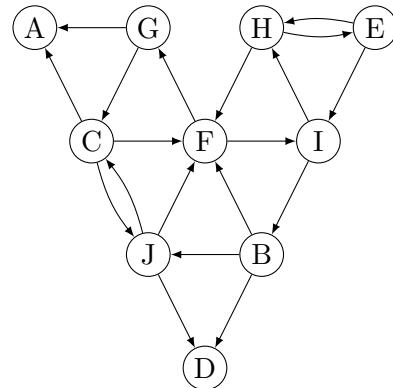
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8

Opgave 212 (Stærke sammenhængskomponenter, 4 %)

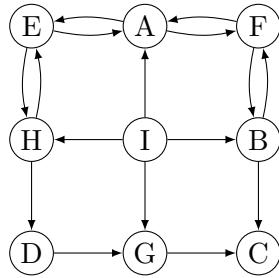
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9 10

Opgave 213 (Stærke sammenhængskomponenter, 4 %)

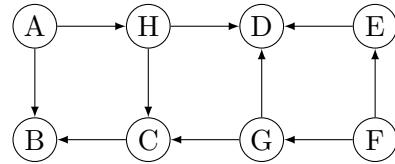
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9 10

Opgave 214 (Stærke sammenhængskomponenter, 4 %)

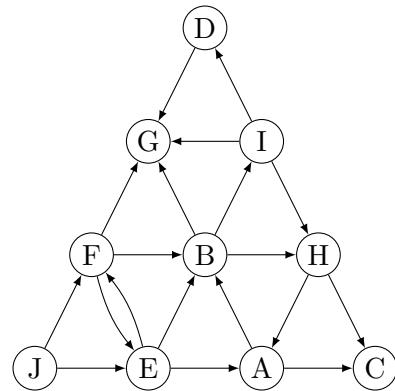
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9

Opgave 215 (Stærke sammenhængskomponenter, 4 %)

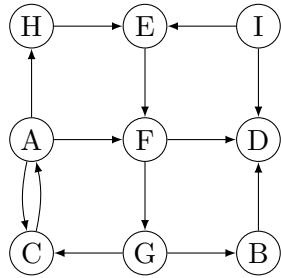
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8

Opgave 216 (Stærke sammenhængskomponenter, 4 %)

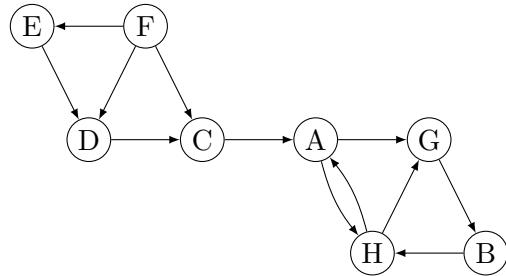
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9 10

Opgave 217 (Stærke sammenhængskomponenter, 4 %)

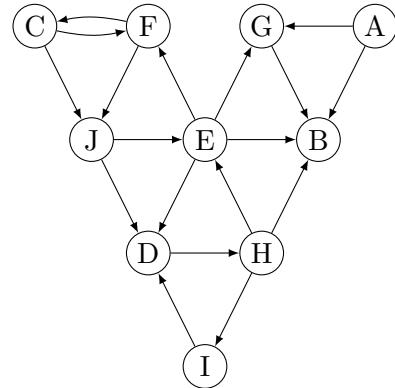
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9

Opgave 218 (Stærke sammenhængskomponenter, 4 %)

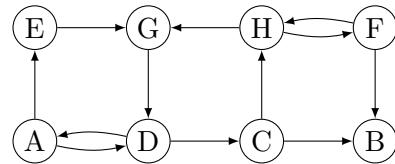
Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8

Opgave 219 (Stærke sammenhængskomponenter, 4 %)

Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8 9 10

Opgave 220 (Stærke sammenhængskomponenter, 4 %)

Hvad er antallet af stærke sammenhængskomponenter i ovenstående graf?

1 2 3 4 5 6 7 8

Opgave 221 (Alle løkke opgaver, 0 %)

Algoritme loop1(n) **Algoritme** loop2(n) **Algoritme** loop3(n)

$i = 1$	$i = 1$	$i = 1$
while $i \leq n$	while $i \leq n$	while $i \leq n$
$j = 1$	$j = 1$	$j = i$
while $j \leq i$	while $j \leq n$	while $j \leq n$
$j = 2 * j$	$j = 2 * j$	$j = 2 * j$
$i = 2 * i$	$i = 2 * i$	$i = 2 * i$

Algoritme loop4(n) **Algoritme** loop5(n) **Algoritme** loop6(n)

$i = n$	$s = 1$	$s = 1$
while $i > 0$	for $i = 1$ to n	for $i = 1$ to n
$j = i$	$j = 1$	$j = s$
while $j > 0$	while $j \leq s$	while $j > 0$
$j = \lfloor j/2 \rfloor$	$j = j + 1$	$s = s + 1$
$i = \lfloor i/2 \rfloor$	$s = 2 * s$	$j = j - 1$

Algoritme loop7(n) **Algoritme** loop8(n) **Algoritme** loop9(n)

$i = 1$	$i = 1$	$i = 1$
$p = 1$	$j = n$	$j = n$
while $p \leq n$	while $i \leq j$	while $i \leq j$
$i = i + 1$	$i = 4 * i$	$i = i * 2$
$p = p * i$	$j = 2 * j$	$j = \lfloor j/2 \rfloor$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\sqrt{n})$ $\Theta(n^3)$ $\Theta(n^2)$ $\Theta(2^n)$ $\Theta(n)$ $\Theta(\log n)$ $\Theta((\log n)^2)$ $\Theta(\frac{\log n}{\log \log n})$ $\Theta(n \log n)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n) **Algoritme** loop2(n) **Algoritme** loop3(n)

$i = 1$
while $i \leq n$
 $i = 2 * i$

$i = 1$
while $i \leq n$
 $i = 3 * i$

$i = 1$
while $i \leq n$
 $i = i + i$

Algoritme loop4(n) **Algoritme** loop5(n) **Algoritme** loop6(n)

$i = 1$
while $i \leq n * n$
 $i = 2 * i$

$i = 1$
while $i \leq n * n$
 $i = 3 * i$

$i = n$
while $i > 0$
if i ulige
 $i = i - 1$
else
 $i = i/2$

Algoritme loop7(n) **Algoritme** loop8(n) **Algoritme** loop9(n)

$s = n$
while $s > 0$
 $s = \lfloor s/2 \rfloor$

$i = 1$
while $i * i \leq n$
 $i = i + i$

$i = 2$
while $i \leq n$
 $i = i * i$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\log \log n)$ $\Theta(n^3)$ $\Theta(n \log n)$ $\Theta(n)$ $\Theta(n^2)$ $\Theta(2^n)$ $\Theta(\sqrt{n})$ $\Theta(\log n)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n) **Algoritme** loop2(n) **Algoritme** loop3(n)

$s = 2$	$i = 0$	$i = 0$
while $s \leq n$	$s = 0$	$s = 0$
$s = s * s$	$q = 0$	while $s \leq n$
	while $q \leq n$	$i = i + 1$
	$i = i + 1$	$s = s + i$
	$s = s + i$	
	$q = q + s$	

Algoritme loop4(n) **Algoritme** loop5(n) **Algoritme** loop6(n)

$i = 1$	$j = n$	$s = 0$
$j = 1$	$i = 1$	$i = 1$
$s = 0$	while $j \geq 0$	while $s \leq n$
while $s \leq n$	$j = j - i$	$s = s + i$
while $j \leq s$	$i = i + 1$	$i = i + 1$
	$j = 2 * j$	
	$s = s + i$	
	$i = i + 1$	

Algoritme loop7(n) **Algoritme** loop8(n) **Algoritme** loop9(n)

$i = 1$	for $i = 1$ to n	$i = 0$
while $i \leq n$	$j = i$	$j = n$
$j = 1$	while $j \leq n$	while $i \leq j$
$k = 1$	$j = 2 * j$	$i = i + 1$
while $k \leq n$		$j = j - 1$
	$j = j + 1$	
	$k = k + j$	
	$i = 2 * i$	

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(\log \log n)$ $\Theta(\sqrt{n} \log n)$ $\Theta(\sqrt[3]{n})$ $\Theta(n)$ $\Theta(n^2)$ $\Theta(\log n)$ $\Theta(\sqrt{n})$ $\Theta(n^3)$ $\Theta(n \log n)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n)	Algoritme loop2(n)	Algoritme loop3(n)
$i = 1$	$i = 1$	$i = 1$
$j = 0$	$j = 1$	$j = 1$
while $i \leq n$	while $i \leq n$	while $i \leq n$
$i = i + i$	while $i \leq n$	while $j \leq i$
while $j < i$	if $i = j$ then	$j = j + 1$
$j = j + 1$	for $k = 1$ to n	$i = 2 * i$
	$s = s + 1$	
	$j = 2 * j$	
	$i = i + 1$	
Algoritme loop4(n)	Algoritme loop5(n)	Algoritme loop6(n)
$i = 1$	$i = 1$	$i = 1$
$s = 0$	$s = 0$	$s = 1$
while $i \leq n$	while $s \leq n$	while $s \leq n * n$
for $j = i$ to n	$j = 1$	$i = i + 1$
$s = s + 1$	while $j \leq i$	$s = s + i$
$i = i + i$	$j = j + 1$	
	$s = s + i$	
	$i = i + 1$	
Algoritme loop7(n)	Algoritme loop8(n)	Algoritme loop9(n)
$i = 1$	$i = 1$	$i = 1$
while $i \leq n$	while $i \leq n$	while $i \leq n$
$j = 0$	$j = 0$	$j = 1$
while $j \leq i$	while $j \leq n$	while $j \leq i$
$j = j + 1$	$j = j + i$	$j = j + 1$
$i = 2 * i$	$i = 2 * i$	$i = 2 * i$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(n \log n)$ $\Theta(2^n)$ $\Theta(\sqrt{n})$ $\Theta(n)$ $\Theta((\log n)^2)$ $\Theta(\log n)$ $\Theta(n^3)$ $\Theta(n^2)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n) **Algoritme** loop2(n) **Algoritme** loop3(n)

$i = n$	$i = n$	$s = 0$
while $i > 0$	while $i \geq 1$	$i = 1$
$i = i - 1$	$j = i$	while $i * i \leq n$
	while $j \leq n$	for $j = 1$ to i
	$j = 2 * j$	$s = s + 1$
	$i = i - 1$	$i = i + 1$

Algoritme loop4(n) **Algoritme** loop5(n) **Algoritme** loop6(n)

$s = 0$	$s = 1$	$s = 1$
$i = n$	for $i = 1$ to n	for $i = n$ to 1 step -1
while $i > 1$	$s = s + 1$	$s = s + 1$
for $j = 1$ to i		
$s = s + 1$		
$i = \lfloor i/2 \rfloor$		

Algoritme loop7(n) **Algoritme** loop8(n) **Algoritme** loop9(n)

$s = 1$	$s = 1$	for $i = 1$ to n
$i = 1$	while $s \leq n$	$j = 0$
while $i \leq n$	$s = s + 1$	while $j \leq n$
for $j = 1$ to i		$j = j + i$
$s = s + 1$		
$i = 2 * i$		

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(n)$ $\Theta(n \log n)$ $\Theta(\sqrt{n})$ $\Theta(n\sqrt{n})$ $\Theta(n^2)$ $\Theta(n^3)$ $\Theta(\log n)$ $\Theta(2^n)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n)	Algoritme loop2(n)	Algoritme loop3(n)
for $i = 1$ to n	for $i = 1$ to n	for $i = 1$ to n
$j = 1$	$j = 1$	$j = i$
while $j \leq i$	while $j \leq n$	while $j > 1$
$j = 2 * j$	$j = 2 * j$	$j = \lfloor j/2 \rfloor$

Algoritme loop4(n)	Algoritme loop5(n)	Algoritme loop6(n)
$i = 0$	$i = 1$	$i = 1$
while $i \leq n$	while $i \leq n$	while $i \leq n$
$j = i$	$j = 0$	$j = 1$
while $j > 0$	while $j \leq n$	while $j \leq i$
$j = \lfloor j/2 \rfloor$	$j = j + 1$	$j = 2 * j$
$i = i + 1$	$i = 2 * i$	$i = i + 1$

Algoritme loop7(n)	Algoritme loop8(n)	Algoritme loop9(n)
$i = 1$	$i = 1$	$s = 0$
while $i \leq n$	while $i \leq n$	$i = n$
$j = i$	$j = n$	while $i > 1$
while $j \leq n$	while $j > 1$	for $j = 1$ to n
$j = j + 1$	$j = j - 1$	$s = s + 1$
$i = 2 * i$	$i = 2 * i$	$i = \lfloor i/2 \rfloor$

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$$\Theta(n^3) \quad \Theta(n) \quad \Theta(\sqrt{n}) \quad \Theta(n \log n) \quad \Theta(\log n) \quad \Theta(n^2) \quad \Theta(n\sqrt{n}) \quad \Theta((\log n)^2)$$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n)

```

for  $i = 0$  to  $n$ 
     $j = 0$ 
     $s = 0$ 
    while  $s \leq i$ 
         $j = j + 1$ 
         $s = s + j$ 

```

Algoritme loop4(n)

```

 $i = 0$ 
 $j = 0$ 
while  $i \leq n$ 
    if  $i < j$  then
         $i = i + 1$ 
    else
         $j = j + 1$ 
     $i = 0$ 

```

Algoritme loop7(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
    for  $j = 1$  to  $n$ 
        if  $i = j$  then
            for  $k = 1$  to  $n$ 
                 $s = s + 1$ 

```

Algoritme loop2(n)

```

for  $i = 1$  to  $n$ 
     $j = 1$ 
    while  $j \leq i$ 
         $j = j + 1$ 

```

Algoritme loop3(n)

```

for  $i = 1$  to  $n$ 
     $j = i$ 
    while  $j > 0$ 
         $j = j - 1$ 

```

Algoritme loop5(n)

```

 $i = 1$ 
while  $i \leq n$ 
     $j = 1$ 
    while  $j \leq i$ 
         $j = j + 1$ 
     $i = i + 1$ 

```

Algoritme loop6(n)

```

 $i = 1$ 
while  $i \leq n$ 
     $j = 1$ 
    while  $j \leq n$ 
         $j = j + 1$ 
     $i = i + 1$ 

```

Algoritme loop8(n)

```

 $s = 0$ 
 $i = n$ 
while  $i > 0$ 
    for  $j = 1$  to  $i$ 
         $s = s + 1$ 
     $i = i - 1$ 

```

Algoritme loop9(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
    for  $j = 1$  to  $i$ 
         $s = s + 1$ 

```

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(n^3)$ $\Theta(n^2)$ $\Theta(\sqrt{n} \log n)$ $\Theta(\log n)$ $\Theta(n)$ $\Theta(\sqrt{n})$ $\Theta(n \log n)$ $\Theta(n\sqrt{n})$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9

Algoritme loop1(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $n$ 
     $s = s + 1$ 
  for  $k = 1$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop4(n)

```

 $s = 1$ 
for  $i = n$  to 1 step  $-1$ 
  for  $j = n$  to 1 step  $-1$ 
     $s = s + 1$ 

```

Algoritme loop7(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $n$ 
    for  $k = 1$  to  $n$ 
       $s = s + 1$ 

```

Algoritme loop2(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
  for  $j = i$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop5(n)

```

for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $i$ 
     $k = 1$ 
    while  $k \leq i + j$ 
       $k = 2 * k$ 

```

Algoritme loop3(n)

```

 $s = 1$ 
for  $i = 1$  to  $n$ 
  for  $j = i$  to  $n$ 
     $s = s + 1$ 

```

Algoritme loop6(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
  for  $j = 1$  to  $i * i$ 
     $s = s + 1$ 

```

Algoritme loop8(n)

```

 $s = 0$ 
for  $i = 1$  to  $n$ 
  for  $j = i$  to  $n$ 
    for  $k = i$  to  $j$ 
       $s = s + 1$ 

```

Algoritme loop9(n)

```

 $s = 0$ 
 $j = 0$ 
for  $i = 1$  to  $n$ 
   $j = j + i$ 
  for  $k = 1$  to  $j$ 
     $s = s + 1$ 

```

Angiv for hver af ovenstående algoritmer udførselstiden som funktion af n i Θ -notation.

$\Theta(2^n)$ $\Theta(n^3)$ $\Theta(\log n)$ $\Theta(n)$ $\Theta(n^2 \cdot \log n)$ $\Theta(\sqrt{n})$ $\Theta(n \log n)$ $\Theta(n^2)$

loop1

loop2

loop3

loop4

loop5

loop6

loop7

loop8

loop9