Danmarks Tekniske Universitet

Skriftlig prøve, den 22. maj 2015.

Kursusnavn: Algoritmer og datastrukturer

Kursusnummer: 02326

Hjælpemidler: Skriftlige hjælpemidler. Det er **ikke** tilladt at medbringe lommeregner.

Varighed: 4 timer.

Vægtning: Opgave 1 - 25 %, Opgave 2 - 25 %, Opgave 3 - 15 %, Opgave 4 - 13 %, Opgave 5 - 22 %. Vægtning er kun approksimativ. Karakteren gives ud fra en helhedsvurdering.

Alle opgaver besvares ved at udfylde de indrettede felter nedenfor. Som opgavebesvarelse afleveres blot denne og de efterfølgende sider i udfyldt stand. Hvis der opstår pladsmangel kan man eventuelt benytte ekstra papir som så vedlægges opgavebesvarelsen.

Asymptotiske grænser skal angives så tætte som muligt. Medmindre andet er angivet er basen på alle logaritmer 2 og $\log^k n$ betegner $(\log n)^k$.

1 Kompleksitet

1.1 (5 %) Angiv for hver af nedenstående udsagn om de er korrekte:

	Ja	Nej
$0,001n + \log n = \Omega(n)$	☐	☐
$2^n + n^4 = O(n^4)$	☐	☐
$\frac{1}{3}\log n + n = \Theta(n)$	☐	☐
$5\sqrt{n} + \log^2 n = O(\sqrt{n})$	☐	☐
$(2n + 6n + 109n^2)n^3 = O(n^4)$	☐	☐

1.2 (5 %) Arranger følgende funktioner i voksende rækkefølge efter asymptotisk vækst. Dvs. hvis funktionen $g(n)$ følger umiddelbart efter funktionen $f(n)$ i din liste skal der gælde at $f(n) = O(g(n))$.

$n^{0.4}$

$2^{1/2 \log n}$

$\frac{1}{4}\log n$

$\frac{1}{2}\log^4 n$

$0,0000003n^3$

$2 \cdot 2^{\log n}$

Svar: ____

1.3 (5 %) Angiv køretiden for nedenstående algoritme. Skriv dit svar i O-notation som funktion af n.

Algoritme 1 Alg1(n)

```
for i = 1 to n do
    j = 1
    while j ≤ n do
        j = j + 2
    end while
end for
```

Svar: ________________________

1.4 (5 %) Angiv køretiden for nedenstående algoritme. Skriv dit svar i O-notation som funktion af n.

Algoritme 2 Alg2(n)

```
for i = 1 to n do
    i = i + 1
end for
j = 1
while j ≤ n do
    j = j + 1
end while
```

Svar: ______________________

1.5 (5 %) Angiv køretiden for nedenstående algoritme. Skriv dit svar i O-notation som funktion af n.

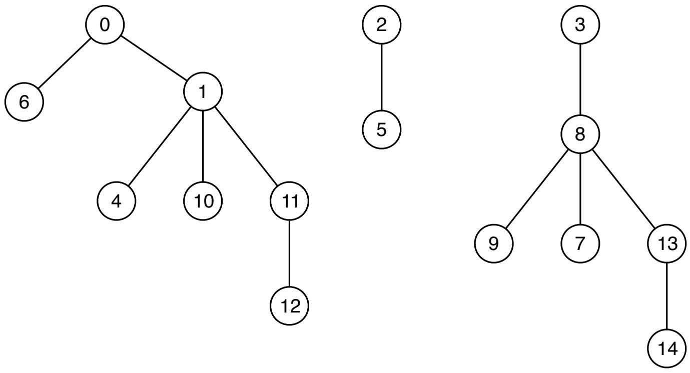
Algoritme 3 Alg3(n)

```
for i = 1 to n do
    j = i
    while j ≤ n do
        j = j · 3
    end while
end for
```

Svar: ______________________

2 Datastrukturer

Betragt følgende skov af træer, der repræsenterer en familie af mængder i en datastruktur til foren og find.



2.1 (5 %) Angiv resultaterne af følgende $\text{FIND}(\cdot)$ operationer.

$\text{FIND}(5)$: __________

$\text{FIND}(6)$: __________

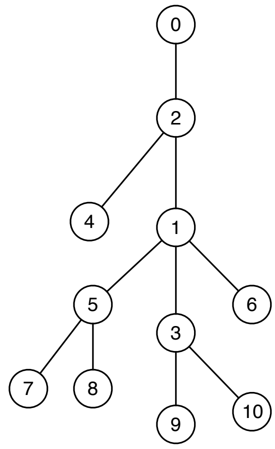
$\text{FIND}(7)$: __________

$\text{FIND}(2)$: __________

2.2 (5 %) Tegn hvordan ovenstående skov af træer ser ud efter operationen $\text{UNION}(4, 9)$ når der anvendes vægtet forening. Antag at $\text{UNION}(i, j)$ altid sætter roden af træet angivet ved i til at være barn af roden af træet angivet ved j når træerne har samme størrelse.

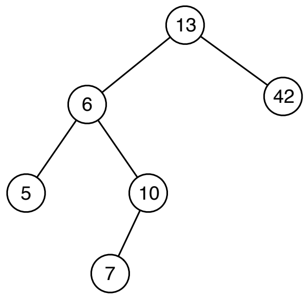
Svar:

2.3 (5 %) Vis resultat af stikompression efter en FIND(3) operation på nedenstående træ.



Svar:

Betragt følgende binære søgetræ.



2.4 (3 %) Angiv rækkefølgen af knuderne i et preorder, postorder og inorder gennemløb af ovenstående binære søgetræ.

PREORDER : ______________________________

POSTORDER : ______________________________

INORDER : ______________________________

2.5 (3 %) Angiv hvordan det ovenstående ovenstående binære søgetræ ser ud efter indsættelse af en knude med nøgle 9.

Svar:

2.6 (4 %) Vi vil gerne understøtte operationen MEMBER(x) på hver af de nedenstående fire datastrukturer. MEMBER(x) returnerer ja hvis værdien x er i datastrukturen og nej ellers. F. eks. hvis datastrukturen indeholder elementerne $\{32, 6, 18, 7, 2\}$ returnerer MEMBER(6) ja og MEMBER(4) nej. Angiv for hver af datastrukturerne hvor lang tid det vil tage at udføre MEMBER(x) operationen i O-notation som funktion af n, hvor n er antallet af elementer i datastrukturen.

Tabel sorteret i stigende rækkefølge: ____________________

Max-hob: ____________________

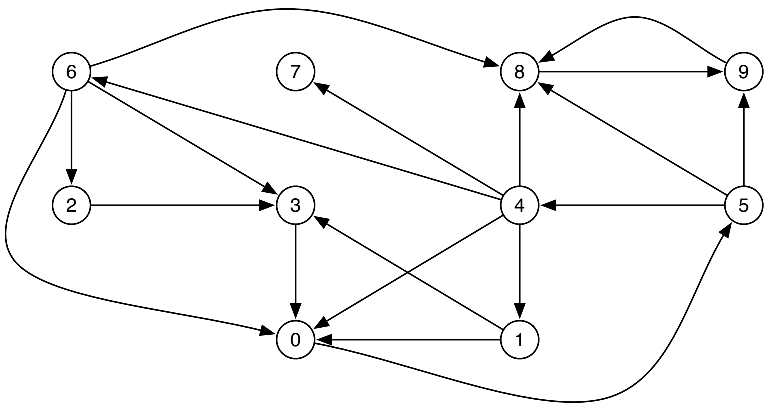
Min-hob: ____________________

Usorteret tabel: ____________________

Binært søgetræ: ____________________

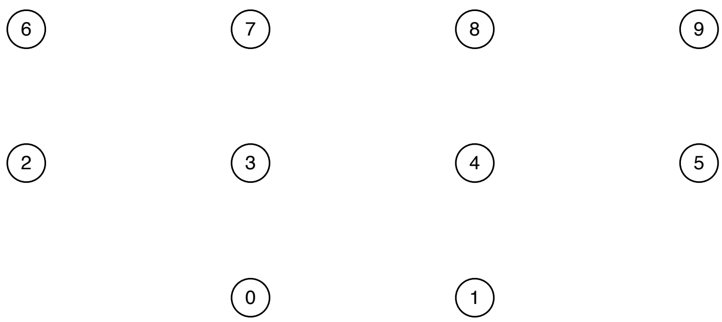
3 Grafer

Betragt følgende graf G.



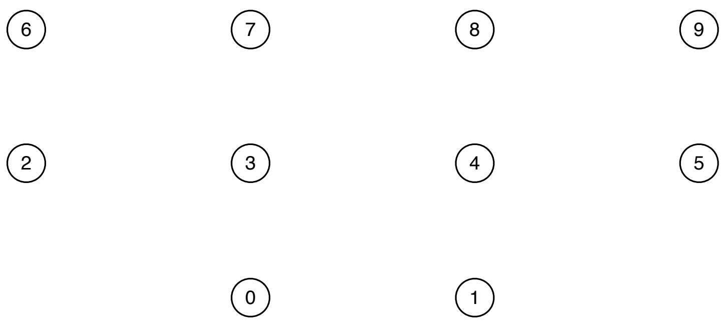
3.1 (5 %) Angiv et DFS træ for G med start i knude 6 og angiv starttid og sluttid for alle knuder. Antag at incidenslisterne er sorteret.

Svar:

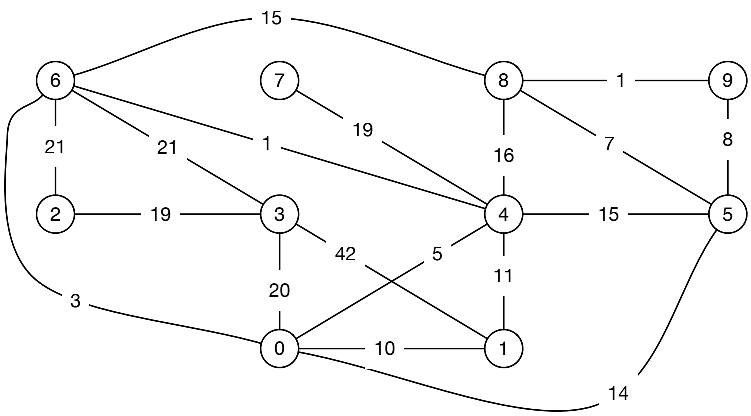


3.2 (5 %) Angiv et BFS træ for G med start i knude 6 og angiv BFS-lag for hver knude. Antag at incidenslisterne er sorteret.

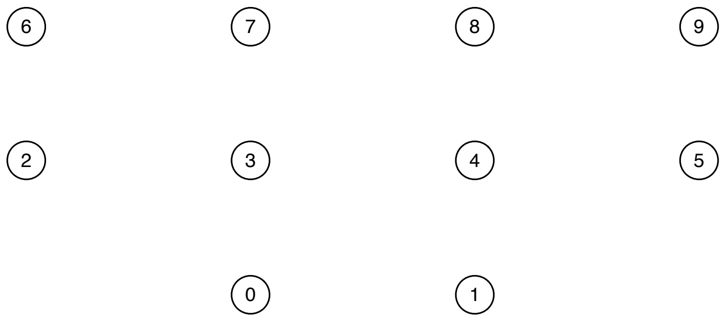
Svar:



3.3 (5 %) Betragt nedenstående graf H. Angiv et mindste udspændende træ T for H samt den samlede vægt af T.



Svar:

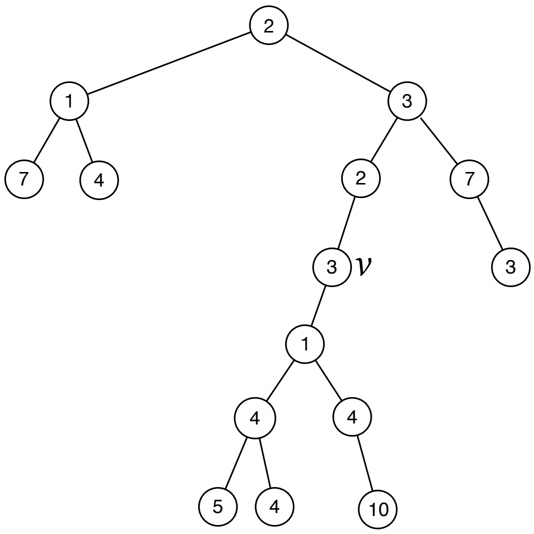


Samlet værdi: ____________

4 Træer

Denne opgave omhandler rodfæstede binære træer. Enhver knude x har felterne $x.parent$, $x.left$ og $x.right$, der betegner hhv. forældre, det venstre barn og det højre barn for x. For roden r, er $r.parent = null$. Derudover har enhver knude x en *vægt* betegnet ved et felt $x.weight$.

4.1 (1 %) Lad x være en knude i træet. *Stisummen* for x er summen af vægtene af alle knuder på stien fra x til roden af træet (inklusive x og r). *Træsummen* for x er summen af vægtene af alle efterkommere af x (inklusive x selv). Betragt nedenstående træ med vægte skrevet i hver knude. Angiv stisum og træsum for knuden v.



Stisum for v : ________________

Træsum for v : ________________

4.2 (5 %) Giv en rekursiv algoritme TRÆSUM(X), der givet en knude x returnerer træsummen for x. Skriv din algoritme i pseudokode og analyser køretiden af din algoritme som funktion af n, hvor n er antallet af knuder i træet.

Svar:

4.3 (7 %) Forklar hvad følgende algoritme gør.

Algoritme 4 TRÆALGORITME(x)

```
if x.parent == null then
    return 1
else
    return 1 + TRÆALGORITME(x.parent)
end if
```

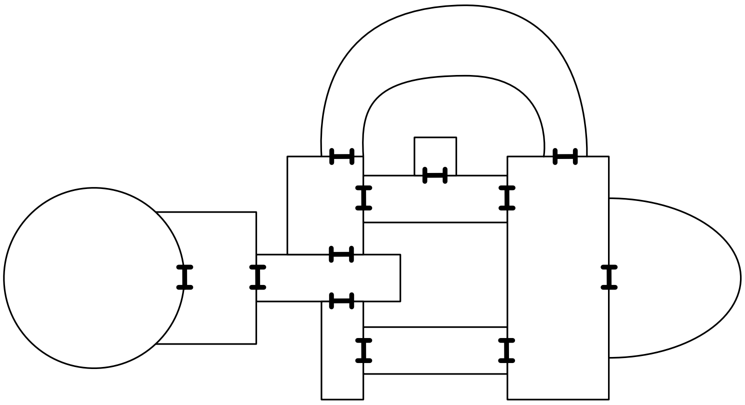
Svar:

Modificer TRÆALGORITME og giv en algoritme STISUM(X), der givet en knude x returnerer stisummen for x. Skriv din algoritme i pseudokode og analyser køretiden af din algoritme som funktion af d, hvor d er dybden af træet.

Svar:

5 Plantegning

Et *plantegning* består af en mængde af R rum $r_0, .., r_{R-1}$ og D døre $d_0, .., d_{D-1}$, der hver forbinder præcis to rum. Hvert rum er en geometrisk figur og en dør mellem to rum er angivet ved tre små fede streger mellem rummene. F. eks. består nedenstående plantegning P af 11 rum og 12 døre.



5.1 (2 %) Beskriv hvordan man kan modellere en plantegning som en graf.

Svar:

5.2 (2 %) Tegn grafen svarende til plantegningen P i eksemplet ovenfor.

Svar:

5.3 (5 %) Vi er nu interesserede i at undersøge om det er muligt at evakuere rummene i tilfælde af en brand. *Entréen* er et særligt rum på plantegningen. En *branddør* er en dør, der automatisk lukker i tilfælde af brand. Et rum kan *evakueres* til entréen hvis der er en forbindelse fra rummet til entreen, der ikke benytter branddøre. Giv en algoritme, der givet en plantegning, en entré e og en mængde B af k branddøre, afgør om alle rum kan evakueres til e. Analyser køretiden af din algoritme som funktion af R, D, og k.

Svar:

5.4 (6 %) Til enhver dør d knytter vi nu et *sikkerhedsniveau*, *sikkerhed*(d), som er et positivt heltal. For at benytte en dør d kræves der at man har sikkerhedsniveau mindst *sikkerhed*(d). Giv en algoritme, der givet en plantegning og to rum r_1 og r_2 finder det minimale sikkerhedsniveau der kræves for at kunne komme fra r_1 til r_2. Analyser køretiden af din algoritme som funktion af R og D.

Svar:

Vi er nu interessede i at udstille kunst i form af enten en skulptur eller et maleri i alle rum på en behagelig facon. En *rute* i plantegning er en sekvens af rum $r_0, \ldots, r_{z-1}$ således at rum r_i og r_{i-1} er forbundet af en dør, og $r_0 = r_{z-1}$. Et rum må gerne besøges mange gange på ruten. En rute er *smuk* hvis der i hvert rum på ruten skiftevis er udstillet en skulptur eller et maleri.

5.5 (2 %) Kig på eksemplet med plantegningen P fra før. Er muligt at udstille en skulptur eller et maleri i hvert rum således at *alle* ruter der starter og slutter i rummet længst til venstre er smukke?

	Ja	Nej
Svar:	☐	☐

5.6 (5 %) Giv en algoritme, der givet en plantegning og en entré e, afgør om alle ruter der starter og slutter i e er smukke. Vi antager at der er en rute fra e til alle rum. Analyser køretiden af din algoritme som funktion af R og D.

Svar: