DANMARKS TEKNISKE UNIVERSITET

Skriftlig prøve, maj 2024

Kursusnavn: Algoritmer og datastrukturer

Kursusnummer: 02326

Hjælpemidler: Skriftlige hjælpemidler er tilladt. Lommeregner er ikke tilladt. Computer er ikke tilladt.

Varighed: 4 timer

Cirka vægtning:

Opgave 1: 25%, Opgave 2: 20%, Opgave 3: 20%,

Opgave 4: 35%.

Vægtningen er kun vejledende da der foretages en afsluttende heldhedsvurdering.

Alle opgaver besvares ved at udfylde eller afkrydse de indrettede svarfelter, svarlinier og svarbokse nedenfor. Som opgavebesvarelse afleveres blot denne og de efterfølgende sider i udfyldt stand. I tilfælde af pladsmangel kan man eventuelt vedlægge ekstra sider.

Medmindre andet er angivet er basen på alle logaritmer 2.

Skriv dit fulde navn og studienummer.

Navn: ______________________________

Studienummer: ______________________________

1 Kompleksitet

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

	Ja	Nej
$2n^3 + n^{10} = O(n^{30})$	☐	☐
$\frac{5(\log n)^3}{\log(n^3)} = \Theta((\log n)^2)$	☐	☐
$45n^5 + n^{\frac{1}{2}} = \Omega(n^5)$	☐	☐
$n \cdot \left(n + n^2 + n^{\frac{1}{3}} + 6\right) = O(n^2)$	☐	☐
$10n^2 + 4n^2(\log(2^n)) = O(n^{2.5})$	☐	☐

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 $f(n) = O(g(n))$.

Angiv funktionens nummer i rækkefølgen (1,2,3,4,5,6) på linien.

$3 \cdot n^3$	________
$\frac{10 \cdot n^3}{\log n}$	________
$42 + \pi$	________
n^n	________
$7 \cdot \sqrt{n} + 8 \cdot \log n$	________
$(0.02 \cdot \log n)^2$	________

1.3 (5%) Angiv køretiden af nedenstående algoritme.

Skriv dit svar i Θ-notation som funktion af n.

ALG1(n)
$k = 0$
for $i = 1$ to $2n - \lfloor n/2 \rfloor$ **do**
  $k = k + 1$
end for
for $j = 1$ to $\lceil n/2 \rceil$ **do**
  $k = k + 1$
end for

Tid: ____________________

1.4 (5%) Angiv køretiden af nedenstående algoritme.

Skriv dit svar i Θ-notation som funktion af n.

ALG2(n)
$i = 1$
while $i \leq n^3$ **do**
  $i = 4 \cdot i$
end while

Tid: ____________________

1.5 (5%) Angiv køretiden af nedenstående algoritme.

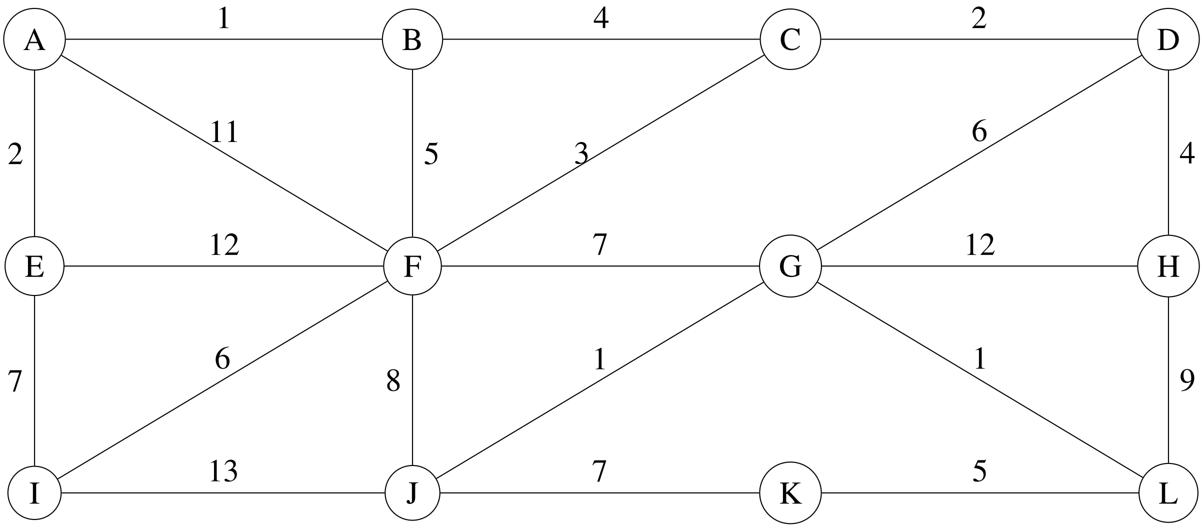
Skriv dit svar i Θ-notation som funktion af n.

ALG3(n)
if $n \leq 1$ **then**
  return 1
else
  return ALG3(⌈n/2⌉) + n^2
end if

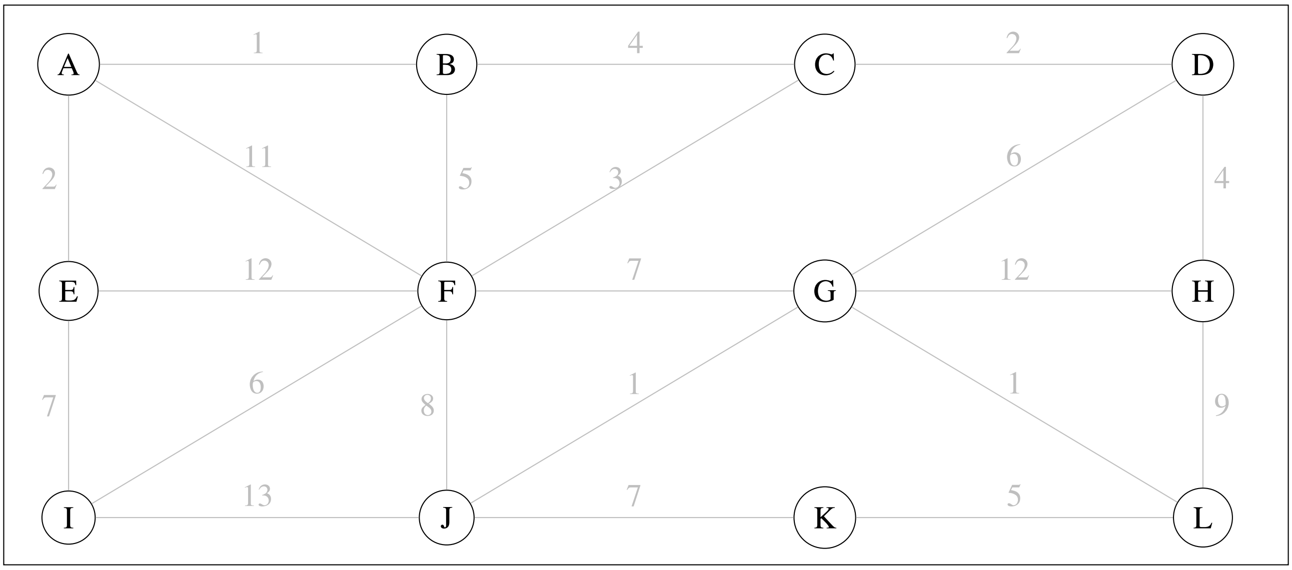
Tid: ____________________

2 Grafer

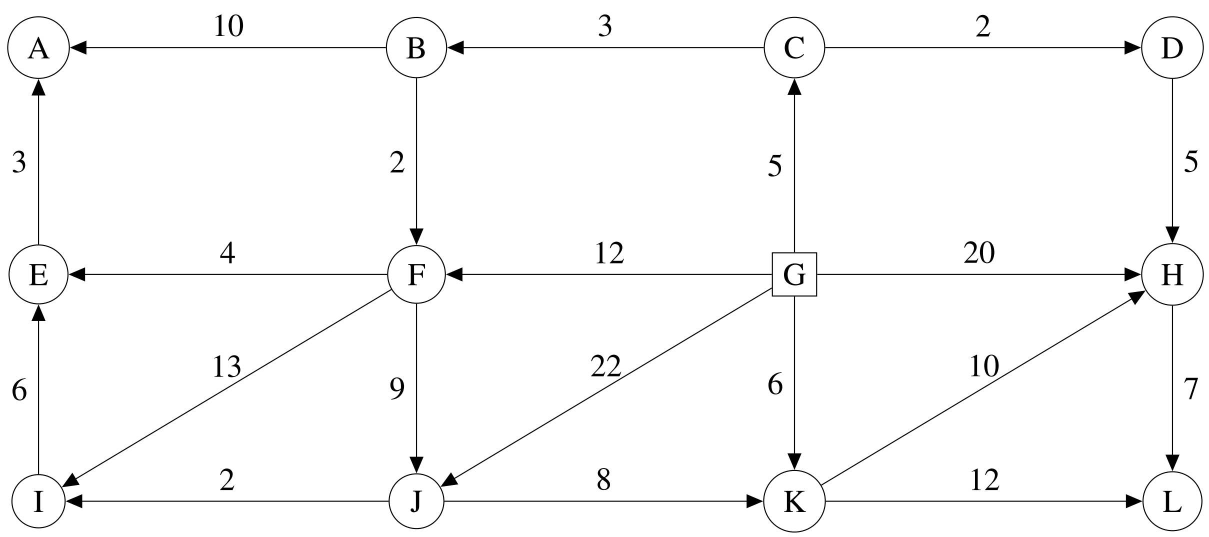
2.1 (5%) Betragt følgende graf:



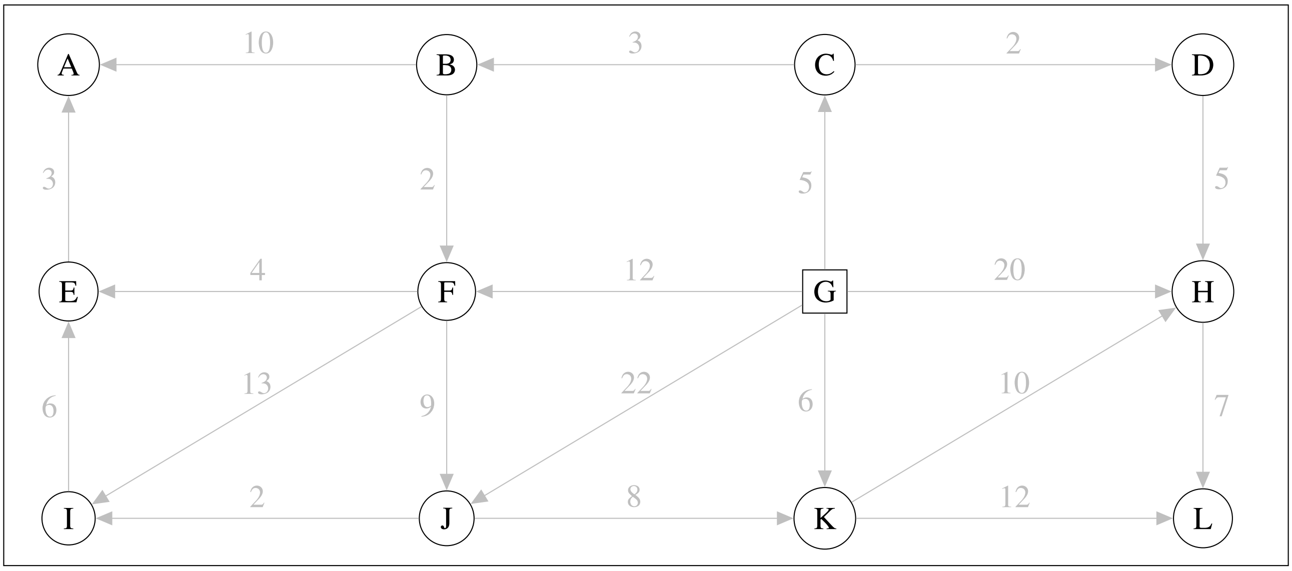
Angiv et mindste udspændende træ nedenfor, ved at markere de kanter, som er med i dit mindste udspændende træ.



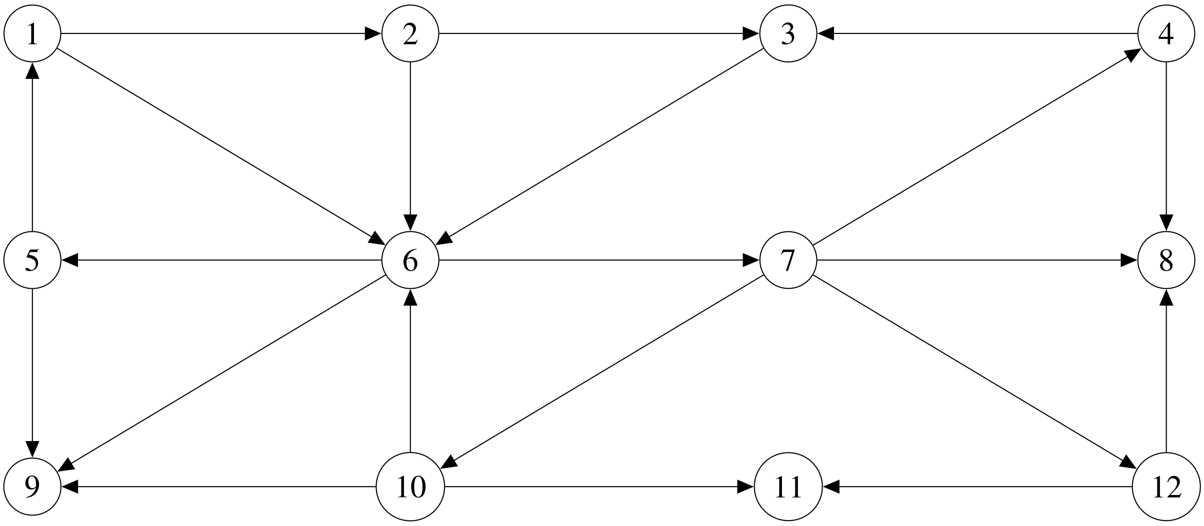
2.2 (5%) Betragt følgende graf:



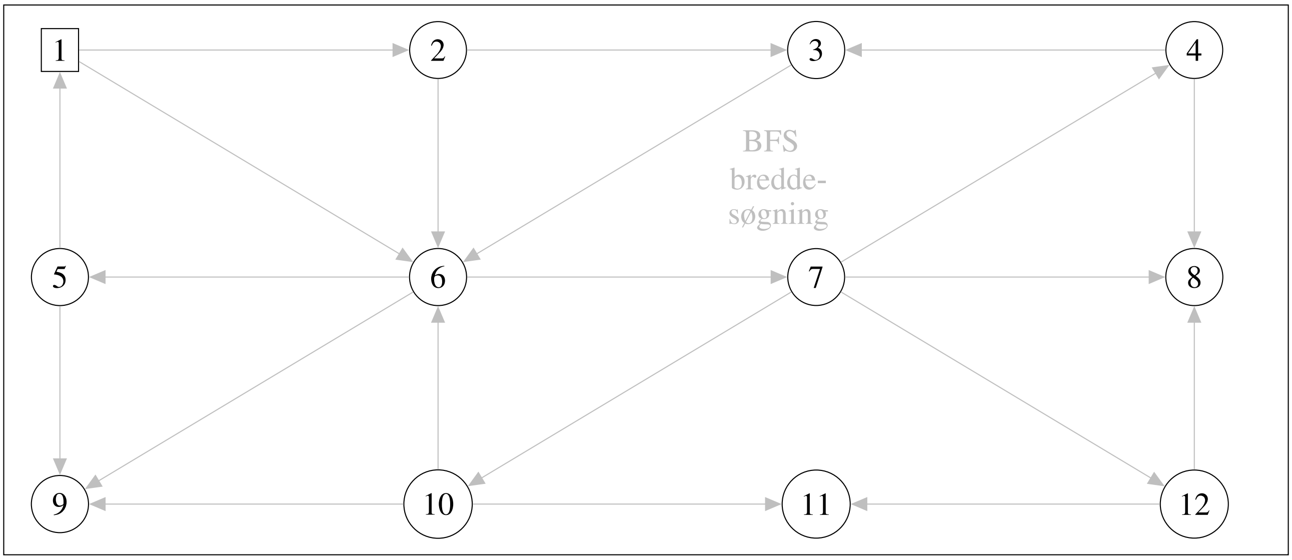
Angiv et korteste-veje-træ fra knuden G nedenfor, ved at markere de kanter, som er med i dit korteste-veje-træ.



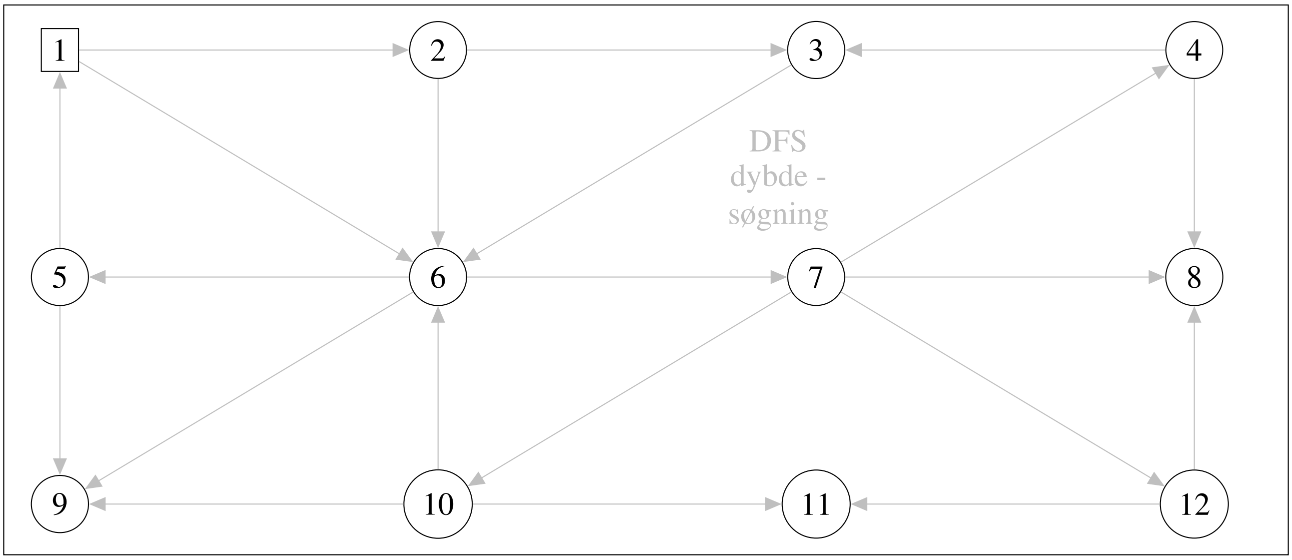
2.3 (10%) Betragt følgende graf:



Kør en bredde-først-søgning fra knuden 1 nedenfor, og marker de kanter, som er med i bredde-først-søgningstræet. Antag, at incidenslisterne er **sorterede**.

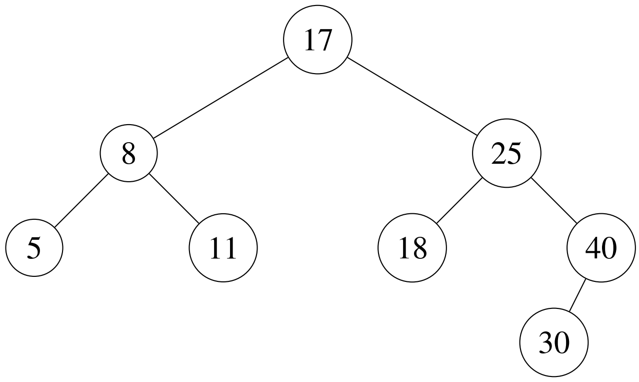


Kør en dybde-først-søgning fra knuden 1 nedenfor, og marker de kanter, som er med i dybde-først-søgningstræet. Antag, at incidenslisterne er **sorterede**.



3 Datastrukturer

3.1 (10%) Betragt følgende binære søgetræ:



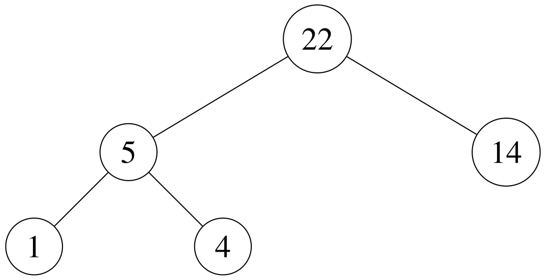
Antag, vi indsætter tallet 19, og dernæst tallet 20.

Tegn det resulterende binære træ.

Antag at vi **i stedet** sletter tallet 25 fra det oprindelige binære træ.

Tegn det resulterende binære træ.

3.2 (10%) Betragt følgende binære max-hob:



Antag, vi indsætter tallet 38.

Tegn den resulterende hob.

og skriv den på arrayform: ______________________________

Antag at vi **dernæst** udfører operationerne `Extract-max` og `Extract-max` igen (altså i alt to gange `Extract-max`).

Tegn den resulterende hob.

og skriv den på arrayform: ______________________________

4 Kabelbaner i Kabiroon

I en galakse langt borte, i landet Kabiroon, er landets $r \cdot c$ byer arrangeret i et $r \times c$ gitter: byerne er placeret i XY-planen så byen (i, j) har x-koordinat i og y-koordinat j, for $i = 1, \ldots r$ og $j = 1 \ldots c$.
Alle byer er navngivet efter deres plads i gitteret, og byen (i, j) er forbundet via en kabelbane med sine naboer til nord, syd, øst og vest, altså $(i + 1, j)$, $(i - 1, j)$, $(i, j + 1)$, $(i, j - 1)$, hvis disse byer findes.
Kabiroon er et bjergland, og hver by b har en højde $h(b)$.
Tiden det tager at rejse fra en by b til dens umiddelbare nabo n med en kabelbane er givet ved 1 plus højdeforskellen, det vil sige: $1 + |h(b) - h(n)|$.
(Husk at $|x|$ betegner den absolutte værdi af x, altså f.eks. $|-5| = 5$.)
Det er også muligt rejse mellem byer som ikke er direkte forbundne; den sammenlagte rejsetid bliver da summen af rejsetiderne langs hver af de direkte forbindelser på rejsen.
Eksempel: $3 \cdot 2$ byer i et 3×2 gitter,

- $(1, 1)$ har højde 1,
- $(2, 1)$ har højde 30,
- $(3, 1)$ har højde 3,
- $(1, 2)$ har højde -1,
- $(2, 2)$ har højde 2,
- $(3, 2)$ har højde 1,

4.1 (10%)

- Beskriv, hvordan byerne og kabelbanerne modelleres som en graf,
- tegn eksemplet ovenfor, og
- angiv den hurtigeste rute fra $(2, 2)$ til $(3, 1)$

4.2 (10%) Vi ønsker nu at anlægge en hovedstad i en af Kabiroons byer.

Givet en by Y er en fjerneste by F en by hvor længden af den hurtigeste vej mellem Y og F er størst mulig.

Vi ønsker at placere hovedstaden C i en by således at vi minimerer afstanden mellem C og den fjerneste by (eller de fjerneste byer) fra C. Vi kalder en sådan by en *central by*.

Betragt først tilfældet hvor $r = 1$, og altså alle byer ligger på en linie.

Giv en algoritme, der finder en central by, når $r = 1$.

Argumenter for korrekthed, og analyser din algoritme som funktion af c og r.

4.3 (15%) Antag nu, at c og r kan tage alle heltalsværdier ≥ 1.

Giv en algoritme, der finder en central placering.

Argumenter for korrekthed, og analyser din algoritme som funktion af c og r.