



KANDIDAT

**470**

PRØVE

# DATS2300 1 Algoritmer og datastrukturer

Emnekode	DATS2300
Vurderingsform	Skriftlig eksamen under tilsyn
Starttid	22.11.2023 14:00
Sluttid	22.11.2023 16:00

---

Sensurfrist11.12.2023 22:59

---

PDF opprettet

11.09.2024 14:27

---

**Informasjon**

Oppgave	Tittel	Oppgavetype
i	Eksamensinformasjon	Informasjon eller ressurser

**Flervalgs**

Oppgave	Tittel	Oppgavetype
1	Kompleksitet	Flervalg
2	Binærtre	Flervalg
3	Tid	Flervalg
4	Huffman-koding	Flervalg
5	Hashfunksjon	Flervalg
6	2-3-4-tre	Flervalg
7	Dijkstra	Flervalg
8	LZW-komprimering	Flervalg

**Langsvar**

Oppgave	Tittel	Oppgavetype
9	Minimumhaug	Langsvar
10	Filtrer LenketListe	Programmering
11	Innsetting/uttagning av minhaug	Muntlig

## 1 Komplexitet

Hva er laveste kompleksitet vi kan få for å fjerne et element fra en kø?

Velg ett alternativ:

☐  $\mathcal{O}(n)$

☒  $\mathcal{O}(1)$

☐  $\mathcal{O}(\log(n))$

☐  $\mathcal{O}(n^2)$

☐  $\mathcal{O}(n \log(n))$

---

Maks poeng: 4

Knytte håndtegninger til denne oppgaven?

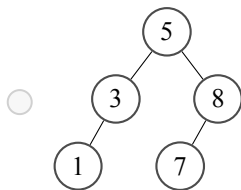
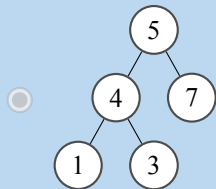
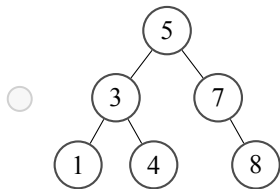
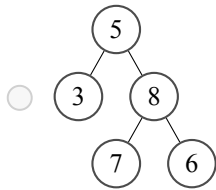
Bruk følgende kode:

**3 2 5 1 4 6 2**

## 2 Binærtre

Hvilket av følgende binærtrær er et komplett binærtre?

Velg ett alternativ:



Maks poeng: 8

Knytte håndtegninger til denne oppgaven?

Bruk følgende kode:

**3 6 9 8 4 9 3**

### 3 Tid

Et program bruker bubblesortering til å sortere en liste, og kan sortere en liste med 1000 elementer på 10 sekunder.

Vi kjører programmet på nytt på en liste med 2000 elementer, og måler hvor lang tid programmet bruker. Hvor lang tid vil programmet bruke?

**Velg ett alternativ:**

☐ Omtrent 70 sekunder

☒ Omtrent 20 sekunder

☐ Omtrent 40 sekunder

☐ Omtrent 15 sekunder

☐ Omtrent 25 sekunder

---

Maks poeng: 8

**Knytte håndtegninger til denne oppgaven?**

Bruk følgende kode:

**6 3 9 7 6 0 6**

## 4 Huffman-koding

Gitt følgende frekvenstabell, lag det kanoniske venstreorienterte Huffmantreet, og komprimer følgende melding: "KJØTTKAKE". Hva blir den kodede meldingen?

Frekvenstabell

A	90
E	152
J	57
K	350
T	179
Ø	34

Velg ett alternativ:

☐ 0111 1111 0100 1000 1100 101

☒ 1000 0000 1011 0111 0011 010

☐ 1001 1001 0011 0111 0001 010

☐ 1010 0010 1000 0001 0111 001

---

Maks poeng: 15

**Knytte håndtegninger til denne oppgaven?**

Bruk følgende kode:

**9 8 1 8 9 1 5**

## 5 Hashfunksjon

Hva er hovedpoenget med en hashtabell?

**Velg ett alternativ:**

- ☐ Lavere minnebruk
- ☐ Sortere data raskere
- ☐ Rask innsetting og henting av verdier
- ☒ Kryptere informasjon

---

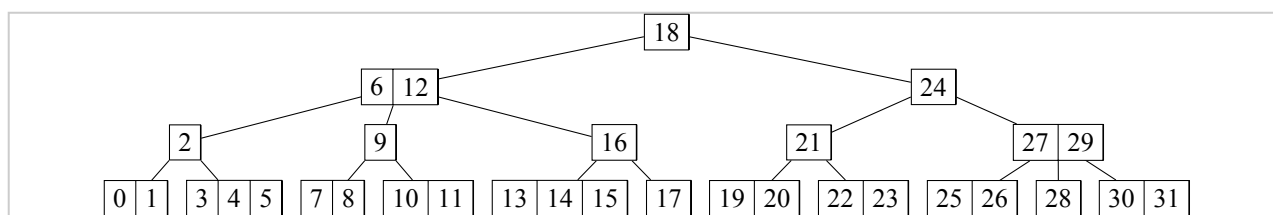
Maks poeng: 6

**Knytte håndtegninger til denne oppgaven?**

Bruk følgende kode:

**2903637**



**6 2-3-4-tre**

Over er det tegnet opp et 2-3-4-tre. Om det gjøres om til et rødsort-tre, hvor mange røde noder vil det være i treet?

Velg ett alternativ:

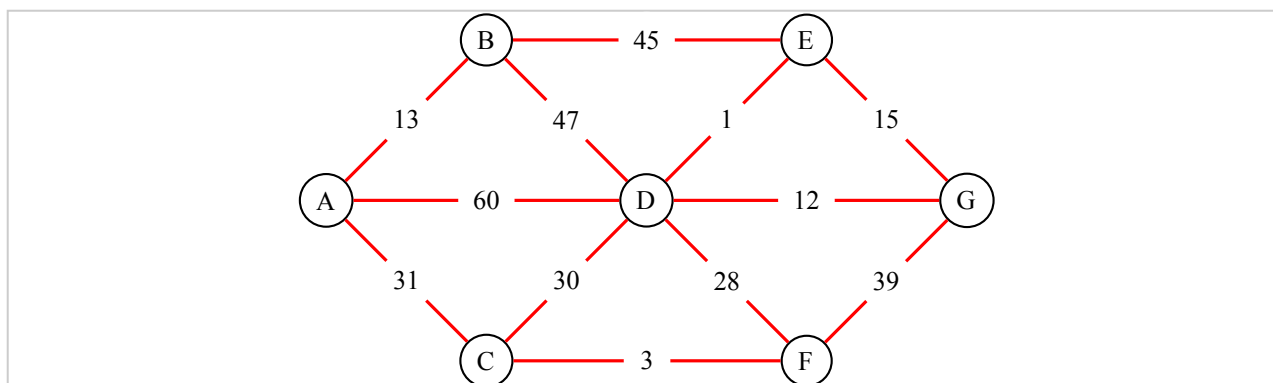
- ☐ 4
- ☐ 21
- ☐ 7
- ☒ 13

Maks poeng: 15

Knytte håndtegninger til denne oppgaven?  
Bruk følgende kode:

**6981301**

## 7 Dijkstra



I Dijkstras algoritme besøkes nodene i en bestemt rekkefølge for å finne korteste vei mellom A og G. Hvilken rekkefølge besøker algoritmen nodene i?

**Velg ett alternativ:**

☐ ABCEFDG

☐ ACBDFEG

☒ ABCFEDG

☐ ABDCEFG

Maks poeng: 15

**Knytte håndtegninger til denne oppgaven?**

Bruk følgende kode:

**0183059**

## 8 LZW-komprimering

Vi skal LZW-komprimere ordet BARBARER, og starter med ordboka med følgende verdier:

A	0
B	1
E	2
R	3

Etter komprimering, hva er den høyeste verdien vi har i ordboka (verdien trenger ikke være brukt i komprimeringa)?

Velg ett alternativ:

☐ 4

☒ 9

☐ 5

☐ 11

Maks poeng: 15

Knytte håndtegninger til denne oppgaven?

Bruk følgende kode:

0 6 2 7 1 7 2

## 9 Minimumhaug

Forklar kort hvorfor det i gjennomsnitt tar konstant tid å legge inn et element i en minimumsheap. Gi en enkel forklaring som viser din forståelse av hvorfor denne operasjonen har konstant tidskompleksitet, uten å gå inn i tekniske detaljer. En til to setninger er nok.

Skriv ditt svar her

Det er fordi den nye noden blir lagt i bunnen, og man sjekker om forelderen har mindre eller lik verdi. i verste fall må alle i grenen trekkes nedover, men i større heaps så gjør man ikke så masse bytter fordi det er så mange noder så det er stor sannsynlighet for at forelderen har samme eller mindre verdi.

Ord: 59

Maks poeng: 10

Knytte håndtegninger til denne oppgaven?

Bruk følgende kode:

0 4 5 3 2 6 1

## 10 Filtrer LenketListe

Denne oppgaven tar utgangspunkt i den minimale LenketListe-koden som er vedlagt.

Du skal lage metoden

```
public LenketListe<T> filtrer(Test<T> p)
```

som gir ut en ny lenket liste som inneholder alle verdiene i den originale lenkede listen som tilfredsstiller `p.test(T verdi)`, i samme rekkefølge som den originale lenkede listen. Den originale lenkede listen skal ikke endres på.

Eksempelvis, om din lenkede liste "lenketListe" består av alle tall fra 1 til 100 i stigende rekkefølge, og testen `p` tester om et tall er mindre enn 50, så skal koden `LenketListe<Integer> nyLenketListe = lenketListe.filtrer(p)` gi deg en ny lenket liste med tallene fra 1 til 49 i stigende rekkefølge. Den originale lenkede listen skal fremdeles da bestå av tall fra 1 til 100, i stigende rekkefølge.

Du kan ikke endre eksisterende kode, men kan lage nye hjelpemetoder om du ønsker.

Prøv å gjør koden så effektiv som mulig, både i tidsbruk og minnebruk. Skriv kommentarer i koden som forklarer hvordan den fungerer, og begrunn eventuelle hjelpestrukturer og hjelpemetoder du har laget.

```
1 public LenketListe<T> filtrer(Test<T> p) {
2     //jeg tolker det slik at oppgaven lar meg filtrere basert på hva som helst
3     //Derfor skal jeg filtrere slik at alle tall(1-100) i den lenkede listen blir p
4     //i tilfelle tom liste
5     if(tom(p)) return;
6     // jeg tolket det også slik at jeg ikke kunne programmere noe i test-interfacer
7     // interfacen måtte blitt gjort inni metoden her derfor setter jeg den lik fals
8     p.test = false;
9     Node<T> gjeldende = p.hode;
10    LenketListe<T> filtrertListe;
11    while(gjeldende != null && gjeldende.neste != null){
12        if(gjeldende.data>0 && gjeldende.data<100){
13            p.test = true;
14            filtrertListe.leggInn(gjeldende.data)
15            gjeldende = p.neste;
16        } else {
17            //slik at det ikke blir uendelig while-loop
18            gjeldende = p.neste;
19        }
20    }
21    //hvis den lenkede listen har null elementer
22    if tom(filtrertListe){
23        System.out.print("Du har ingen tall mellom 1 og 100 i din lenkede liste")
24        return;
25    }
26    //metoden her har O(n) i kompleksitet som er noe dårlig men det er fordi den må
27    //problemet med denne metoden er at det kan potensielt komme flere duplikater a
28    slik går bra
29    return filtrertListe;
30 }
```

Maks poeng: 30

**Knytte håndtegninger til denne  
oppgaven?**

Bruk følgende kode:

**6 6 5 4 9 2 1**

## 11 Innsetting/uttagning av minhaug

Vi skal lage en minimumsheap ved å sette inn heltall sortert ved naturlig ordning.

Tegn hvordan treet ser ut etter å ha satt inn tallene 7, 23, og 19, i denne rekkefølgen.

Tegn hvordan det ser ut etter at du har lagt inn først 17 og så 2 i det samme treet.

Tegn hvordan treet ser ut etter at du har lagt inn først 3 og så 13 i det samme treet.

Forklar hvordan du legger inn tallet 5 i det samme treet, og tegn hvordan treet ser ut etter denne innleggingen.

Fjern minste element fra treet, og tegn treet etter denne fjerninga. Forklar hvordan du fjernet elementet.

---

Maks poeng: 30

**Knytte håndtegninger til denne oppgaven?**

Bruk følgende kode:

**0805883**







OSLOMET

**i** Fill out Question Code and Test Information on every sheet. Fyll inn oppgavekode og emneinformasjon på alle skisseark.

Total number of pages

2

Totalt antall ark

(Only on the first sheet/Kun på første side)

Question Code

Oppgavekode

Date

Dato

Subject code

Emnekode

Candidate ID

KandidatID

Question nr

Oppgavenr

Page number

Sidetall

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

22/11/23

DATS2300

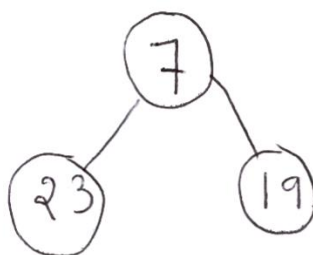
470

11

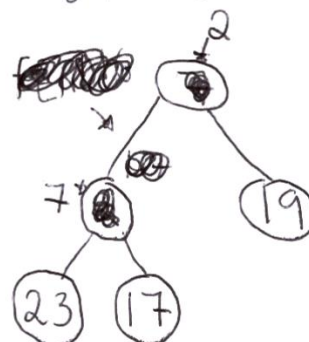
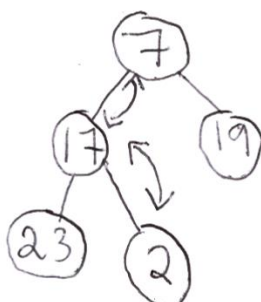
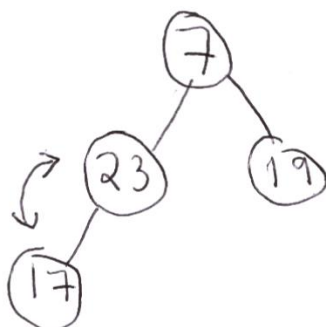
1

Drawing area Tegneområde

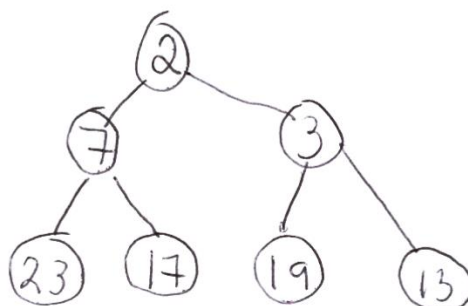
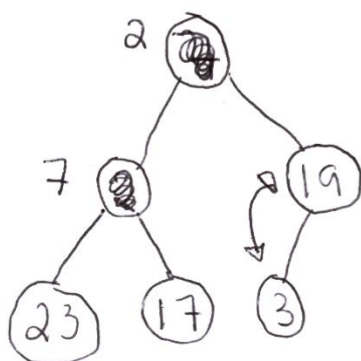
TRINN 1 [7, 23, 19]



TRINN 2 Legge inn 17 så 2



TRINN 3 Legge inn 3 så 13



Use only black or blue pen / Benytt kun svart eller blå penn





**i** Fill out Question Code and Test Information on every sheet. Fyll inn oppgavekode og emneinformasjon på alle skisseark.

**OSLOMET**

Total number of pages  
Totalt antall ark  
(Only on the first sheet/Kun på første side)

Question Code  
Oppgavekode

Date  
Dato

Subject code  
Emnekode

Candidate ID  
KandidatID

Question nr  
Oppgavenr

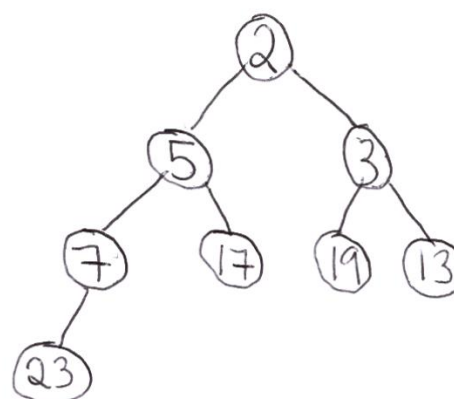
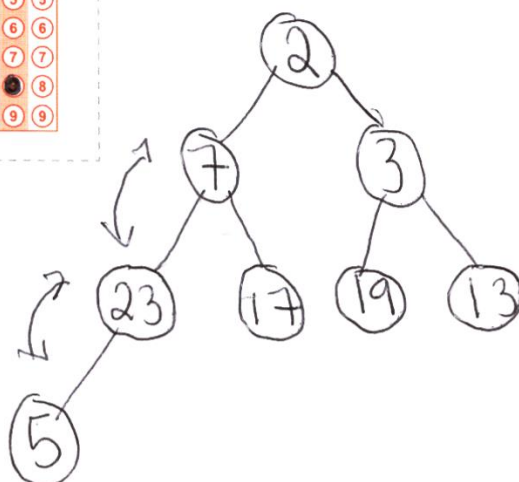
Page number  
Sidetall

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

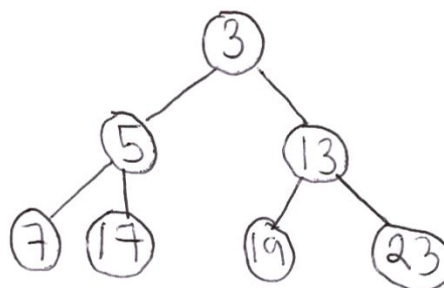
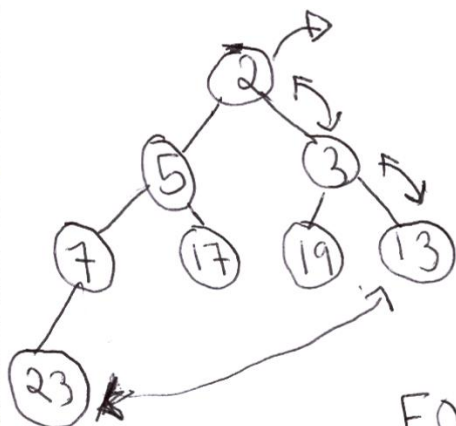
22/11/23	DATS2300	470	11	2
----------	----------	-----	----	---

**Drawing area** Tegneområde

TRINN 4 LEGG INN 5



TRINN 5 fjern minste



**FORKLARING =**

- I min-heap så fjerner man rotelemtet(2)
- også må man hente opp det minste barnet(3)
- Man må også passe på at det blir komplett
- Så man flytter opp (23)

↓  
fyllt inn fra  
venstre i siste  
nivå

Use only black or blue pen / Benytt kun svart eller blå penn