

## Eksamen i 45011 Algoritmer og Datastrukturer

Torsdag 12. januar 1995, Kl. 0900-1300.

**Hjelpemidler:** Godkjent lommekalkulator samt alle trykte og håndskrevne hjelpemidler.

**Faglig kontakt:** Bjørn Olstad, tlf.: 73593447

---

### Oppgave 1 (6 %)

I et sanntidsinstrument trenger man å få sortert tabeller med 10.000 reelle tall. En ny slik tabell ankommer hvert sekund og det er derfor avgjørende at sorteringen alltid kan gjøres på under ett sekund. For å velge riktig sorteringsalgoritme har man implementert en del sorteringsalgoritmer og tatt tiden på sortering av en slik vektor med 10.000 elementer:

Sorteringsmetode	Tid i millisekunder
Innsettingssortering	16230 ms
Boblesortering	22160 ms
Quicksort	450 ms
Heapsort	610 ms
Mergesort	850 ms

Foreslå hvilken av sorteringsalgoritmene i tabellen ovenfor som bør velges. Begrunn svaret.

### Oppgave 2 (6 %)

Nedenfor er det vist to alternative måter å implementere QuickSort på:

Alternativ A:

```
QuickSortA( A,p,r )
if (r-p)≥8
    then  q ← Partition( A,p,r )
          QuickSortA( A,p,q )
          QuickSortA( A,q+1,r )
    else  Innsettsorter( A,p,r )
```

Alternativ B:

```
QuickSortB( A,p,r )
if (r-p)≥8
    then  q ← Partition( A,p,r )
          QuickSortB( A,p,q )
          QuickSortB( A,q+1,r )
```

.....  
QuickSortA( A,1,n )

.....  
QuickSortB( A,1,n )  
Innsettsorter( A,1,n )  
.....

Forklar kort hvorfor det kan være nyttig å benytte Innsettsorter i QuickSort.

Hva tror du er mest tidseffektivt av alternativ A og alternativ B gitt ovenfor. Begrunn valget.

### Oppgave 3 (6 %)

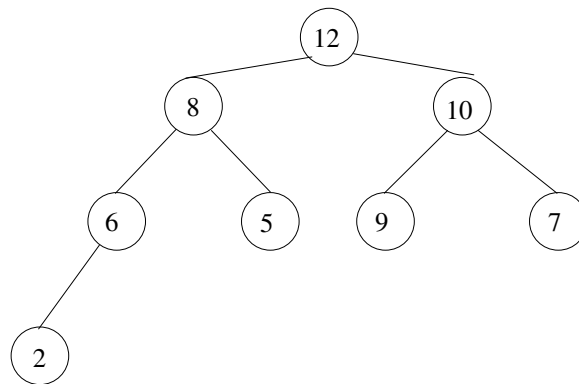
Det skal lages et program som gjør raskt oppslag i en samling med 10.000.000 poster. Postene kan sorteres etter den nøkkelen som det skal søkes på. Selve dataene utgjør 600 MB og dataene skal legges på et eksternt medium som harddisk, CD-ROM, etc. Programmet skal kunne kjøres på en personlig datamaskin med kun 4 MB arbeidslager (RAM). Gi et begrunnet valg for enten et 23-tre eller et generelt B-tre som datastruktur for de 10.000.000 postene i følgende to tilfeller:

**3a)** Det eksterne mediumet er kjennetegnet ved svært treg søketid (ved tilfeldig posisjonering), men svært rask dataoverføringsrate.

**3b)** Det eksterne mediumet er kjennetegnet ved svært rask søketid (ved tilfeldig posisjonering), men svært lav dataoverføringsrate.

### Oppgave 4 (6 %)

I figuren under er det vist en haug (heap). Utfør 3 **Extract-Max** operasjoner på den viste haugen og sett deretter inn tallet 20. Tegn haugen slik den vil se ut etter disse 4 operasjonene.



### Oppgave 5 (6 %)

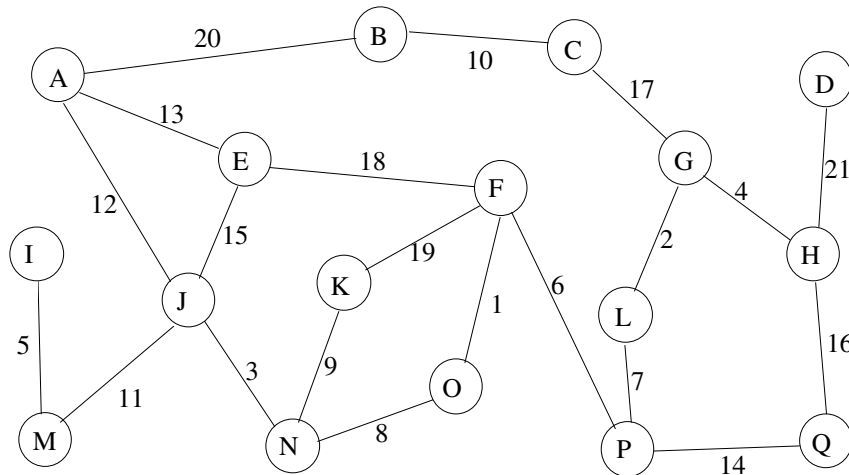
Det er i pensum vist at sortering er  $\Omega(n \log n)$ . Samtidig vises det at tellesortering kan være  $O(n)$ . Forklar kort hvorfor disse to utsagnene ikke motsier hverandre.

### Oppgave 6 (6 %)

Forklar kort formålet med en hash-funksjon. Tegn deretter hash-tabellen man får etter at man har satt inn heltallene 5, 23, 37, 44, 33 og 87 i en hash-tabell med hash-funksjonen:  $h(k) = k \bmod 10$ . Tallene skal settes inn i nevnte rekkefølge og kollisjoner skal løses ved hjelp av "linear probing".

## Oppgave 7 (30 %)

I figuren nedenfor er det vist en urettet graf med kostnader på kantene:



- 7a) Gi en lovlig nummerering av nodene "A" ..., "Q" ut fra et BFS-søk som starter i node "A".
- 7b) Gi en lovlig nummerering av nodene "A" ..., "Q" ut fra et DFS-søk som starter i node "A".
- 7c) Gi et enkelt bevis for at kanten mellom node "N" og "O" hører med i grafens minimale spennentre.
- 7d) Finn grafens minimale spennentre ved hjelp av Prims algoritme. Tegn det resulterende spennntreet og merk kantene med et nummer som indikerer i hvilken rekkefølge Prims algoritme har innlemmet kantene i det spennntreet som bygges opp av algoritmen.
- 7e) Kjør Dijkstras algoritme på grafen med node "A" som kildenode. Fyll inn korteste avstand til node "A" i alle nodene.
- 7f) Finn den maksimale flyten man kan ha mellom node "A" og node "D".

## Oppgave 8 (6 %)

Et selskap har samlet alle sine elektroniske dokumenter og tilgjengelige oppslagsverk i en ustrukturert "tekstfil" som utgjør 500 MB. En bruker har lyst til å søke i denne tekstfila etter strengen:

"kjente feil med produktet QuickJazz"

Hvilken tekstsøkealgoritme tror du er best egnet til å utføre dette søket. Du kan anta at hele tekstfila ligger i maskinens arbeidslager. Begrunn valget.

## Oppgave 9 (28 %)

**9a)** Vi skal reise langs en fast reiserute med bil. Langs denne ruta kjenner vi til hvor det er bensinstasjoner og hvilken pris hver enkelt bensinstasjon tar for den bensintypen som vår bil benytter. Oppgaven går ut på å finne ut hvilke bensinstasjoner vi bør tanke på samt hvor mange liter vi bør tanke på hver av disse bensinstasjonene. Målet er å bruke minst mulig penger for å kunne nå fram til bestemmelsesstedet.

Vi kan anta at tanken er tom når vi starter og at tanken maksimalt rommer 40 liter bensin. Det spiller ingen rolle hvor mye bensin som er på tanken når vi kommer fram til bestemmelsesstedet. Videre vil vi anta at bilen bruker 1 liter bensin per mil uansett kjøremåte og veiforhold.

Beskriv en best mulig algoritme for dette problemet i pseudo-kode og angi algoritmens tidskompleksitet som funksjon av  $N$  som er antall bensinstasjoner langs den valgte reiseruta. Du kan anta at algoritmens inndata er gitt ved:

VAR

Avstand : Array[1..N] of Integer;  
Pris : Array[1..N] of Real;

"Avstand[i]" er avstanden i antall mil fra startsstedet og fram til bensinstasjon nummer "i" langs den valgte reiseruta, mens "Pris[i]" er bensinprisen på denne bensinstasjonen. Startsstedet og bestemmelsesstedet er gitt ved henholdsvis bensinstasjon nummer 1 og  $N$ .

**9b)** I denne oppgaven skal vi løse det samme problemet som i oppgave (9a), men reiseruta er nå ikke fastlagt på forhånd. Du kan gjøre de samme antagelsene som i oppgave (9a).

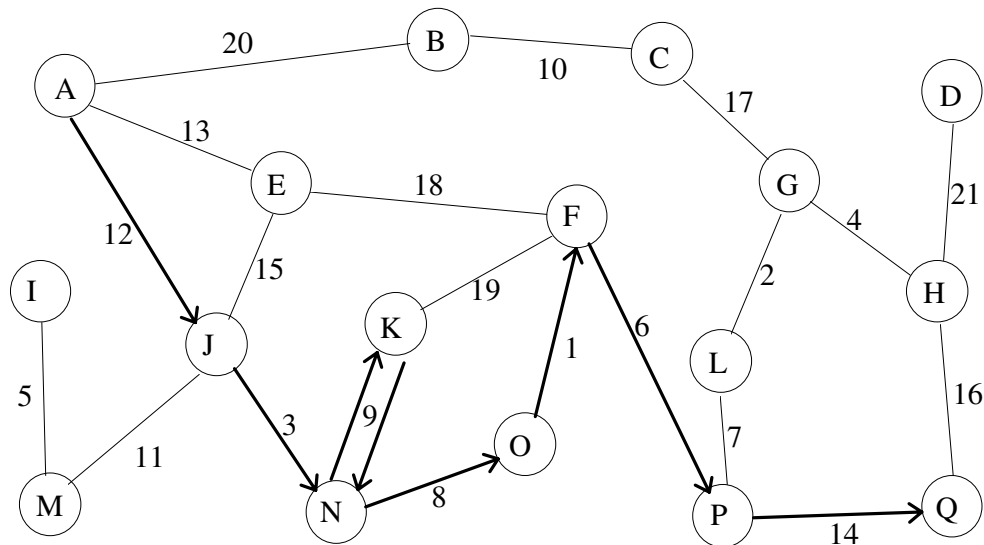
Beskriv en best mulig algoritme i pseudo-kode som finner en reiserute fra startsstedet og fram til bestemmelsesstedet. Reiseruta skal være slik at det koster minst mulig å nå fram til bestemmelsesstedet langs denne ruta. Du kan anta at algoritmens inndata er gitt ved:

VAR

Avstand : Array[1..N, 1..N] of Integer;  
Pris : Array[1..N] of Real;

"Avstand[i,j]" er avstanden i antall mil fra bensinstasjon nummer "i" og fram til bensinstasjon nummer "j".  
Avstand[i,j]= $\infty$  hvis det ikke går en direkte vei mellom bensinstasjonene uten andre bensinstasjoner. "Pris[i]" er bensinprisen på bensinstasjon nummer "i". Startsstedet og bestemmelsesstedet er gitt ved henholdsvis bensinstasjon nummer 1 og  $N$ .

**HINT:** En mulig løsning av problemet er gitt ved Dijkstra algoritme anvendt på en graf som er avledet av inndataene til problemet og parametrene som er gitt i antagelsene under oppgave (9a). Oppgaven kan besvares ved å beskrive denne grafen (hvordan er nodene, kantene og kantkostnadene definert?). På neste side er det gitt et praktisk eksempel som illustrerer problemet.

Eksempel for oppgave (9b):

Bensinstasjon "1" = node "A" (Startsstedet)

Bensinstasjon "2" = node "B"

..

Bensinstasjon "17" = Node "Q" (Bestemmelsesstedet)

Avstand[i,j] er angitt med kanter og kostnader i figuren. Ingen kanter er tegnet der  $\text{Avstand}[i,j]=\infty$ . Den sorte veien viser optimal reiserute og tabellen under gir bensinprisen på stasjonene og hvor mye man burde fylle på de ulike stasjonene:

Bensinstasjon	Pris (kr/liter)	Tanking i liter
A	20.00	12
B	25.00	
C	10.00	
D	6.50	
E	18.00	
F	17.50	
G	12.00	
H	8.00	
I	12.00	
J	17.00	12
K	4.00	24
L	4.00	
M	16.00	
N	18.00	
O	25.00	
P	3.50	14
Q	3.00	

I dette tilfellet er altså beste reiserute "A" - "J" - "N" - "K" - "N" - "F" - "P" - "Q" og den minste prisen for denne turen er  $(20 \cdot 12 + 17 \cdot 12 + 4 \cdot 24 + 3.50 \cdot 14)$  kroner = 589 kroner.