



Institutt for datateknikk
og informasjonsvitenskap

Eksamensoppgave i

TDT4225 Lagring og behandling av store datamengder

Fredag 2. desember 2011, kl. 0900-1300

Oppgaven er utarbeidet av faglærer Kjell Bratbergsengen og kvalitetssikrer Svein-Olaf Hvasshovd

Kontaktperson under eksamen: Kjell Bratbergsengen, telefon 7359 3439 og 906 17 185

Språkform: Bokmål

Tillatte hjelpemidler: D

Ingen trykte eller håndskrevne hjelpemiddel er tillatt.

Bestemt, enkel kalkulator tillatt.

Sensur: 23. desember 2011

Oppgave 1, Videoserver (10%)

En videoserver skal levere film til enkeltkunder hvor hver videostrøm i gjennomsnitt krever 3 MB/sekund.

Disker er koplet til en fiberoptisk buss med maksimal kapasitet 200 MB/s.

Serveren har en intern buss med maksimal kapasitet 400 MB/s.

Ingen enheter skal belastes mer enn 70% av full kapasitet.

Disker gjør 10000 rotasjoner per minutt og gjennomsnittlig sporlengde er 800 KB.

Serverens arbeidslager er 8 GB.

Hvor mange kunder kan én server betjene?

Vis hvordan du kommer fram til resultatet og presiser forutsetningene.

Forslag til løsning

Vi må kontrollere om det er busser eller disk som er flaskehals. Arbeidslageret er så stort at vi kan bruke store blokker. 1, 2 eller flere spor i en blokk.

Kontroll av buss:

Data skal fra disk til buffer, det er 2 passeringer. Det samme gjelder når data skal til nettet, da må også data to ganger over bussen.

Maksimal datastrøm over bussen: $400 * 0,7/4 = 70 \text{ MB/s}$

Disken klarer maksimalt $V=0,7*0,8*10000/60=93,3$ MB/s.

Dette er en kapasitet som ikke kan oppnås, disken vil måtte skifte posisjon mellom blokker. Regner en rotasjon i tap per blokk.

#spor	93,33
1	46,67
2	62,22
3	70,00
4	74,67
5	77,78

I tabellen er det regnet ut kapasitet som funksjon av antall spor per blokk: $V_{eff} = \frac{V_{max} \times s}{s+1}$
s er antall spor per blokk.

Koplingen mellom disk og server er mer enn rask nok.

Maksimalkapasiteten begrenses av bussen. Blokk lengden blir 3 spor og kapasiteten er 70 MB/s .
En server klarer $70/3=23$ samtidige klienter.

Oppgave 2, R-tre for lagring av to-dimensjonale objekter (20%)

- Forklar oppbyggingen av en R-tre-fil.
- Tabellen viser en indeksblokk i et R-tre.

Koordinater				
x0	y0	x1	y1	Blokk
7	1	11	3	A
10	7	15	20	B
5	4	6	17	C
15	5	20	6	D
4	4	16	9	E

Du skal søke etter et objekt med koordinat $\langle 8,13 \rangle$. Hvilke subtrær må søkes gjennom for å finne objektet.

- Indeksblokken må splittes. Bruk kvadsplitt. Hvilke omhyllingsrektangler blir kimer i de to nye blokkene?
- Ingen blokk skal ha færre enn to indekser. Fullfør splittingen av indeksblokken.

Forslag til løsning

a) Se lærebok.

b) Koordinaten ligger utenfor omhyllingsrektanglene til *alle* subtrær. Dette finnes lettest ved å tegne opp rektanglene. Dermed skal ingen subtrær besøkes.

c) Må regne ut omhyllingsarealet av 10 kombinasjoner: AB, AC, AD, AE, BC, BD, BE, CD, CE og DE

Koordinater						A	B	C	D	E	AD	CB
x0						7	10	5	15	4	7	5
y0						1	7	4	5	4	1	4
	x1					11	15	6	20	16	20	15
		y1				3	20	17	6	9	6	20
7	1	11	3	A		8	152	96	65	96	65	190
10	7	15	20	B		152	65	160	150	192	247	160
5	4	6	17	C		96	160	13	195	156	240	160
15	5	20	6	D		65	150	195	5	80	65	240
4	4	16	9	E		96	192	156	80	60	128	192
7	1	20	6	AD		65	247	240	65	128	65	285
5	4	15	20	CB		190	160	160	240	192	285	160

Dette er satt opp i regneark.

Formelen er: $A_{i,j} = (\max(x_1^i, x_1^j) - \min(x_0^i, x_0^j)) \times (\max(y_1^i, y_1^j) - \min(y_0^i, y_0^j))$

Kombinasjonen CD gir størst omhyllingsareal, de er da kimer i hver sin blokk.

d) Splittingen fullføres ved å legge objekter til den kime som utvider arealet minst. Objektene gjennomgås i den rekkefølge de er lagret.

A legges til D, samlet areal 65, utvidelse er 60.

B legges til C, samlet areal 160, utvidelse er $160-13=147$.

B lagt til AD vil gi et samlet areal på 247, en utvidelse på $247-65=182$ som er mye større.

Da har begge blokker fått sitt minimum, den siste blokken E legges til der det utvides minst.

EAD gir 128. Utvidelse er $128-65=63$.

EBC gir 192. Utvidelse er $192-160=32$.

E legges til BC, minst utvidelse av areal.

Nye blokker er: **AD** og **BCE**.

Oppgave 3, Sortering (15%)

a) Du skal gjøre initiell sortering av følgende poster med reservoarmetoden, reservoaret har plass for 4 poster.

Innfilens nøkler ser slik ut: 32,44,10,3, 5,79,64,43,98,33,8,3,5,2.

Vis resultatet ved å skrive ut alle delfilene.

b) Du har fått 370 delfiler og kan flette maksimalt 90 delfiler. Hvor mange dymmy delfiler må du legge til i første fletting for å få et optimalt flettetre?

c) Alle delfilene har volum 1. Hva blir samlet I/O-volum for hele fletteprosessen?

Forslag til løsning

a)

Reservoar	Delfil 1	
32,44,10, <u>3</u>	3	
32,44,10, <u>5</u>	5	
32,44, <u>10</u> ,79	10	
<u>32</u> ,44,64,79	32	
<u>43</u> ,44,64,79	43	
<u>98</u> ,44,64,79	44	
98, <u>33</u> ,64,79	64	
98,33, <u>8</u> ,79	79	
<u>98</u> ,33,8,3	98	Delfil 2
5,33,8, <u>3</u>	3	
<u>5</u> ,33,8,2	5	
-,33, <u>8</u> ,2	8	
-, <u>33</u> ,-,2	33	Delfil 3
-,,-, <u>2</u>	2	

Den gjennomsnittlige delfillengde er $2M$ når postene kommer i tilfeldig rekkefølge. M er størrelsen på reservoaret.

b) Flettegrad $p = 90$, antall delfiler $N=370$.

$$\text{Antall fletterunder er } y = \left\lceil \frac{N}{p-1} \right\rceil = \left\lceil \frac{370}{90-1} \right\rceil = 5$$

$$N+x-y(p-1)=1$$

$$370+x-5*(90-1)=1, \Rightarrow x=1-370+5*89=76$$

Antall dummyfiler i første fletting er 76.

c) Flettevolum: De 4 første flettinger fletter: $(90-76=14)+90+90+90=296$ delfiler

Den siste flettingen fletter alle delfiler = 370.

Satt opp i regneark:

p	x				pass	vol
90	76	p	p	p	r1	
	14	90	90	90	284	568
					370	740
					654	1308

Sum passeringer av delfiler er 654, hver passering innebærer 1 lesing og 1 skriving.

Volumet blir 1308 delfilenheter.

Oppgave 4, Lagring av matriser. (20%)

Et system for lagring av store matriser lagrer matrisen som kvadratiske submatriser. Hver submatrise lagres som en blokk på disk, 16 KB stor. Hvert element i matrisen krever 4 byte.

a) Hvor stor i antall elementer blir hver submatrise?

b) Systemet brukes til å lagre to matriser, $R(2000,3000)$ og $S(3000,4000)$. Hvor mye plass på disken opptar de to matrisene? Tallene i parentes angir antall linjer henholdsvis kolonner i matrisen).

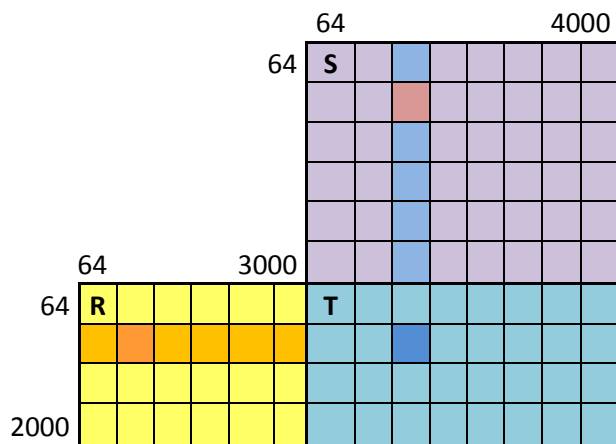
c) Matrisene skal multipliseres: $T=R*S$. Skisser algoritmen du vil bruke for multiplikasjonen. Hvor stor blir T ?

d) Anta at primærlageret bare har plass til tre submatriser. Hvor stort blir I/O-volumet for matrisemultiplikasjonen?

e) Hvis du har plass for 100 submatriser i arbeidslageret, hva blir nå I/O-volumet. Begrunn svaret.

Forslag til løsning

a) Sidekanten i den kvadratiske submatrisen er 64 elementer. Kvadratroten av 4K elementer.



b) $R(2000,3000)$ og $S(3000,4000)$.

Antall rader med submatriser i R blir $\lceil \frac{2000}{64} \rceil = 32$.

Antall kolonner med submatriser i R blir $\lceil \frac{3000}{64} \rceil = 47$.

Antall rader med submatriser i S blir $\lceil \frac{3000}{64} \rceil = 47$.

Antall kolonner med submatriser i S blir $\lceil \frac{4000}{64} \rceil = 63$.

Volumet av R er $32 \times 47 \times 16384 = 24641536$ byte = 24,6 MB.

Tilsvarende utregning for S gir volum av S: 48,5 MB.

Summen av de to matrisene er 73,1 MB.

```
c) for l = (1,1,32) do{
  for k = (1,1,63) do{
    for j=(1,1,47) do {
      read (r(l,j);  read(s(j,k);
      t(l,k) = t(l,k) + r(l,j)×s(j,k);    // Obs: matrisemultiplikasjon
    }
    write t(l,k);
  }
}
```

T(2000,4000)

d) Antall rader med submatriser i R blir $L = \left\lceil \frac{2000}{64} \right\rceil = 32$.

Antall kolonner med submatriser i R blir $M = \left\lceil \frac{3000}{64} \right\rceil = 47$.

Antall rader med submatriser i S blir $M = \left\lceil \frac{3000}{64} \right\rceil = 47$.

Antall kolonner med submatriser i S blir $N = \left\lceil \frac{4000}{64} \right\rceil = 63$.

s er volumet av submatrisen.

Formelen blir $V = LN(2M+1)s = 32*63*(2*47+1) * 16 \text{ KB} = 3,13 \text{ GB}$

e) Alternativ 1:

Plass for 100 submatriser. Hver rad består av 47 submatriser, en rad og en kolonne er inne når en submatrise blir fullført. Raden kan da brukes om igjen for alle kolonner. Siden alle submatrisene i en rad får gjenoppfrisket prioriteten sin for hver submatrise som lages i resultatmatrisen vil ikke matrisene bli skjøvet ut av LRU før en ny rad blir behandlet. En ny rad blir lest inn under dannelsen av den første submatrisen på en ny submatriserad. .

Formelen blir nå: $32*(47+63*47+63) * 16 \text{ KB} = 1,61 \text{ GB}$.

Alternativ 2:

En kan lese inn to rader med submatriser fra R, det er 94 submatriser. En holder to submatriser fra resultatmatrisen T i lager. Så leses en kolonne med submatriser fra S og en danner danner 2 submatriser i T.

Algoritmen blir:

```
for l = 1, l+2 while l < L do {
  for j = (1, 1, M) do { read (r(l, j)); if l < M then read (r(l+1, j)); };
  for k = (1, 1, N) do {
    for j = (1, 1, M) do {
      read(s(j, k));
      t(l, k) = t(l, k) + r(l, j) * s(j, k);
      if l < M then
        t(l+1, k) = t(l+1, k) + r(l+1, j) * s(j, k);
    }
    write t(l, k);
    if l < M then write t(l+1, k);
  }
}
```

$V = sL(2M+2+MN)/2 = L(M+1+MN/2)s = 32*(47+1+47*63/2)*16 \text{ KB} = 801,4 \text{ MB}$

Alternativ 3:

Endrer algoritmen og multipliserer superblokker. Hver superblokk blir på 25 submatriser, 5 ganger 5 matriser i hver.

Antall rader med supersubmatriser i R blir $\left\lceil \frac{32}{5} \right\rceil = 7$.

Antall kolonner med supersubmatriser i R blir $\left\lceil \frac{47}{5} \right\rceil = 10$.

Antall rader med supersubmatriser i S blir $\left\lceil \frac{47}{5} \right\rceil = 10$.

Antall kolonner med supersubmatriser i S blir $\left\lceil \frac{63}{5} \right\rceil = 13$

Formelen for volumet blir nå: $V=7*13*(2*10+1)*25*16 \text{ KB}=782,7 \text{ MB}$

Alle alternativer vil hver for seg gi full uttelling på eksamen.

Oppgave 5, Relasjonsalgebra (25%)

- a) Forklar relasjonsalgebraoperasjonen snitt: $\mathbf{R} = \mathbf{A} \cap \mathbf{B}$. Hvilke krav stilles til operandene A og B.
- b) Forklar hvordan snitt blir utført med gjentatte gjennomløp. Finn også et generelt uttrykk for operasjonens totale transportvolum når en bruker gjentatte gjennomløp.
- c) Tabellen A(a₁, a₂, a₃, a₄) har 3 millioner poster a 500 byte. A1 er nøkkel med en gjennomsnittlig lengde på 15 byte.
Tabellen B(b₁, b₂, b₃, b₄) har 1 million poster a 500 byte. B1 er nøkkel med gjennomsnittlig nøkkellengde 15 byte. Hva blir operasjonens I/O-volum? Tilgjengelig reservoar er 10 MB.
- d) For begge tabeller eksisterer det en lagret signaturtabell på primærnøkkelen. Signaturtabellene er enklest mulig og er uten postreferanse. En signatur tar 4 byte.
Forklar hvordan du vil utføre snittoperasjonen nå.
- e) Hva blir I/O-volumet for den modifiserte snitt-algoritmen?

Forslag til løsning

- a) Nøklene må i det minste være sammenlignbare. Posten bør være like da vi skal velge fra hvilken tabell vi velger resultatpost.
- b) Den minste operands nøkkel stilles opp i arbeidslager. Hvis overløp, leses 2. operand på nytt inntil hele førsteoperand er lest. En trenger bare nøkkelen i arbeidslager fordi resultatposten tas fra B-operanden. En formel for I/O-volumet er $V=V_a+V_bV_{ak}/\alpha M+V_r$
- c) I oppgaven er B den minste operanden, derfor er det Bs nøkler som skal stilles opp i M.

	Antall poster	Postlengde	Nøkkellengde	Signatur
A	3000000	500	15	4
	byte	1500000000	45000000	12000000
	MB	1500	45	12
B	1000000	500	15	4
	byte	500000000	15000000	4000000
	MB	500	15	4

Bs nøkler har et volum på er 15 MB, to gjennomløp må til.

B leses en gang, A leses to ganger.

$V=500 + 2*1500\text{MB}=3500\text{MB} = 3,5 \text{ GB}$ pluss volumet av R.

Hvis en legger hele poster i WS vil gjentatte gjennomløp gi et mye større volum:
 $V = 500 + 1500 * 500/10 = 75,5$ GB pluss volumet av R.

d) Les tabellen med signaturer fra B og la dem ligge i WS. Alle signaturer får plass.
Les signaturer fra A. Slett alle signaturer i WS som ikke har maken signatur i A.
Les B, nøkler som har treff på signatur i WS lagres i WS.
Les A, nøkler som har treff på signatur i WS testes mot lagret nøkkel, hvis nøkler er like så lagres A-posten i R.

Siden signaturtabellen ikke inneholder postreferanser har vi ingen mulighet til å unngå lesing av hele operandtabeller, - det gjelder både A og B.

e) I/O volumet blir: Signaturtabeller og basetabeller leses alle en gang:
 $V=4+12+500+1500=2016$ MB = 2,0 GB
Volumene for signaturtabellene er vist i regnearket over.

Oppgave 6, Optimalisering (10%)

a) Hvor mange forskjellig algebra –tre er det mulig å få til når en forener tre tabeller?

b) Gitt:

```
SELECT A.x,B.s FROM A,B,C WHERE  
      A.x>21 AND A.x=C.y AND C.y<B.s AND B.s <100;
```

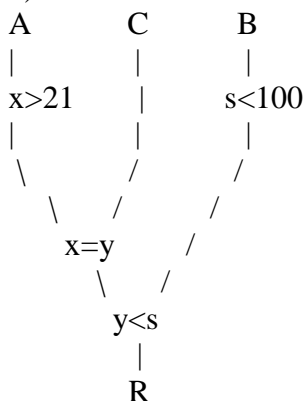
Tegn opp algebra-treet for utførelsen av uttrykket slik det står.

c) Kan uttrykket omskrives slik at utførelsen blir mer effektiv? Tegn opp det modifiserte algebra-treet.

Forslag til løsning

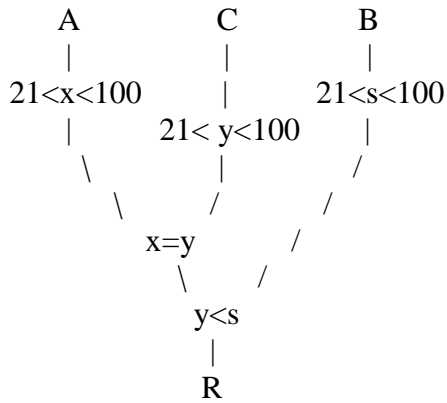
a) Antall operandsekvenser er $3!=1*2*3=6$.
Antall operatortre er 2.
Totalt antall varianter blir $6*2=12$.

b)



Det er viktig å skyve alle seleksjoner så nær basetabellene som mulig for å redusere volume. På samme måte bør vi også sette inn projeksjoner for å ta bort attribute som ikke inngår i resultatet eller i prosessen fram til resultatet.

c)



Uttrykket kan skrives om med sikte på å overføre seleksjon på en basetabell til andre basetabeller. Den enkleste overføring skyldes likhetsforening mellom A og C. $x > 21$ og ulikheten kan direkte overføres til y i C. Vi har en generell forening mellom AC og B. $y < s$ som i B skal være mindre enn 100. Dermed kan denne seleksjonen overføres både til A og C.