

Institutt for datateknikk og informatikk

Eksamensoppgave i TDT4150 Avanserte databasesystemer

Faglig kontakt under eksamen: Jon Olav Hauglid

Tlf.: 93 80 58 51

Eksamensdato: Fredag 9. juni 2017

Eksamenstid (fra-til): 9.00 – 13.00

Hjelpemiddelkode/Tillatte hjelpemidler: D. Ingen trykte eller håndskrevne hjelpemidler tillatt. Bestemt, enkel kalkulator tillatt.

Annen informasjon: Oppgavesettet inneholder 6 oppgaver. Det er angitt i prosent hvor mye hver (del-)oppgave teller ved sensur. Gjør rimelige antagelser der du mener oppgaveteksten er ufullstendig og skriv kort hva du antar. Lykke til!

Målform/språk: Bokmål

Antall sider (uten forside): 2

Antall sider vedlegg: 0

Informasjon om trykking av eksamensoppgave

Originalen er:

1-sidig ☐ **2-sidig** ☒

sort/hvit ☒ **farger** ☐

skal ha flervalgskjema ☐

Kontrollert av:

23.05.17 Svein Erik Bratsberg (sign)

Dato

Sign

Oppgave 1 – Arkitektur (10 %)

- «Relational Query Processor» og «Transactional Storage Manager» er to viktige komponenter i et relasjonsdatabasesystem. Forklar kort hva hver av disse to komponentene gjør.
- Anta at du skal lage en sky-løsning for databaser. Hvilken parallell arkitektur ville du valgt og hvorfor? («Shared Memory», «Shared Disk», «Shared Nothing»).

Oppgave 2 – NoSQL og LSM-trær (20 %)

- Forklar «consistent hashing» og hvordan det kombineres med replikering av data (project Voldemort).
- Hva er dokument-orientert lagring (MongoDB) og hvordan påvirker det skjemamodellering? Gi et eksempel på hvordan Prosjekter og Deltagere («Workers») kan relateres til hverandre i dokument-orientert lagring.
- Forklar hvordan regioner («regions») og regionservere i Apache HBase fungerer når mengden av data øker.
- Forklar begrepet «rolling merge» som er brukt i LSM-trær («log-structured merge trees»).

Oppgave 3 – Spørreoptimalisering (30 %)

- Anta at join skal utføres på tabellene A og B. Tabell A har T_A tupler og tabell B har T_B tupler. Hvor kostbart vil det være å gjøre dette med Nested Loop Join? Hva med Merge Join? Beskriv eventuelle antagelser du gjør.
- Det kan være vanskelig for en spørreoptimalisator å behandle en spørring der tre tabeller joines på samme måte som en spørring der 30 tabeller skal joines. Forklar hvorfor og forklar kort hva som må gjøres forskjellig.
- En blokkerende relasjonsoperator er en operator som krever at en eller flere operander er fullstendige før operatoren kan utføres (dvs. kan ikke utføres på delresultat). Gi to eksempler på ikke-blokkerende operatore og to eksempler på blokkerende operatore.
- Nevn minst tre grunner til at en spørring kan ta lang tid å utføre.

Oppgave 4 – Distribuerte databasesystemer (15 %)

- Hva er replikering og hvorfor kan det være ønskelig?
- En transaksjon ønsker å først oppdatere rad X og deretter rad Y. En annen transaksjon ønsker å oppdatere rad Y først og deretter rad X. Forklar hvorfor dette kan gi vranglås ved synkron gruppereplikering. Hva skjer i det samme tilfellet hvis asynkron gruppereplikering blir brukt?

Oppgave 5 – Rangering og skyline (10 %)

- Anta en tabell med hybler som inneholder pris og avstand til Gløshaugen. Lag et eksempel som viser at Skyline-operatoren brukt på denne tabellen kan gi mer enn to hybler som svar.
- Krever Rank Join at begge operandene er sortert først? Begrunn svaret.

Oppgave 6 – Tema fra seminarartikler (15 %)

- a) Fungerer SharedDBs løsning for spørreoptimalisering best ved høy eller lav last? Begrunn svaret.
- b) Forklar kort hvordan NoDB klarer å aksessere data i rådatafiler effektivt.
- c) Anta en tabell T med en kolonne C der tuplene i utgangspunktet er i tilfeldig rekkefølge. Ved bruk av Database Cracking, hva skjer hvis det kommer en spørring med betingelsen «T.C > 10»? Hvis det etterpå kommer en spørring med «T.C < 8» og en spørring med «T.C > 15», hva skjer hvis det til slutt kommer en spørring med «T.C > 8 AND T.C < 12»?