



KANDIDATNUMMER:

EKSAMEN

EMNENAVN: Databasemodellering og databasesystemer

EMNENUMMER: IMT2571

EKSAMENS DATO: 14.12.2012

KLASSE: 10HBSPA, 11HBDRA, 11HBIDATA,
11HBISA, 11HBPUA

TID: 9.00–14.00

EMNEANSVARLIG: Rune Hjelsvold (918 60 974)

ANTALL SIDER UTLEVERT: 10

TILLATTE HJELPEMIDLER: Ingen

INNFØRING MED PENN.

Ved innlevering skilles hvit og gul besvarelse og legges i hvert sitt omslag.

Oppgavetekst, kladd og blå kopi beholder kandidaten.

Husk kandidatnummer på alle ark.

Les hele oppgavesettet før du begynner og planlegg hvordan du bør bruke tiden for best å besvare hele settet. Beskriv antagelser og fortolkninger du gjør i besvarelsen dersom oppgaveteksten synes uklar.

Oppgave 1 – SQL og relasjonsdatabaser (30%)

Gitt disse tabellene:

Leverandør

lId	lNavn	klasse	lBy
L1	MPX	1	Sandefjord
L2	Deal	1	Oslo
L3	Harddisk	2	Bergen

Komponent

kId	kNavn	vektGr	prNavn
K1	DDR3 32GBRAM	NULL	Corsair
K2	DDR3 16GBRAM	90	Kingston
K3	GeForce GTX 670 2GB PhysX CUDA	1100	Gainward
K4	PCI Gigabit Ethernet Adapter	300	D-Link

Prosjekt

pId	pNavn	pBy
P1	CloudPC	Gjøvik
P2	ByteEater	Oslo
P3	SimuMonster	Bergen

LKP

lId	kId	pId	antall
L1	K2	P1	50
L1	K4	P1	30
L2	K1	P2	400
L2	K1	P3	20
L3	K3	P3	20
L3	K4	P3	20

a) (10%)

Vis resultatet når følgende spørringer kjøres – illustrer gjerne hvordan du kommer fram til resultatet:

- ```
SELECT DISTINCT lNavn, pNavn
FROM Leverandør l, Prosjekt p, LKP
WHERE LKP.lId = l.lId AND LKP.pId = p.pId
ORDER BY pNavn, lNavn
```
- ```
SELECT kNavn, prNavn, SUM(antall)
FROM Komponent
JOIN LKP USING (kId)
GROUP BY kNavn, prNavn
```
- ```
SELECT lNavn
FROM Leverandør
WHERE lId IN (
 SELECT lId
 FROM (
 SELECT DISTINCT lId, pId
 FROM LKP) AS LP
 GROUP BY lId
 HAVING COUNT(*) > 1)
```

#### b) (10%)

Lag SQL-setninger som:

- Lister ut navn på alle leverandører som har levert RAM (anta at "RAM" forekommer i komponentnavnet) til noen av prosjektene. Spørringen skal også liste ut navn på produsenten av RAM-brikkene som ble levert. Resultatet skal være sortert alfabetisk på leverandøres navn og deretter på produsentens navn. Kjørt på databasen gitt på forrige side skal spørringen gi dette resultatet

| <i>lNavn</i> | <i>prNavn</i> |
|--------------|---------------|
| Deal         | Corsair       |
| MPX          | Kingston      |

2. Henter ut ei liste over prosjektnavn og by for alle prosjekter. Lista skal også inneholde navnet på eventuelle leverandører som finnes i samme by som prosjektene. Kjørt på databasen gitt ovenfor skal spørringen gi dette resultatet

| <i>pNavn</i> | <i>pBy</i> | <i>lNavn</i> |
|--------------|------------|--------------|
| CloudPC      | Gjøvik     | NULL         |
| ByteEater    | Oslo       | Deal         |
| SimuMonster  | Bergen     | Harddisk     |

**c) (10%)**

Hva mener vi med begrepet integritet i en relasjonsdatabasesammenheng? Gi eksempler på typiske problemer som kan oppstå dersom man ikke har integritetskontroll i databasehåndteringssystemet. Hvilke mekanismer tilbyr et relasjonsdatabasehåndteringssystem for integritetskontroll?

**Oppgave 2 – Databasesdesign (35%)**

**a) (20%)**

IB-linja på videregående skole er en del av en internasjonal utdanning kalt International Baccalaureate Diploma Programme. I følge statistikk fra IB World School er det ca. 2.400 videregående skoler over hele verden som tilbyr IB-linjer med over 100.000 elever totalt. IB-elever over hele verden har samme pensum, samme eksamen og felles eksamenstider.

Anta at du er hyret inn av IB-ledelsen internasjonalt til å utvikle en spørsmålsdatabase som kan brukes for testing av elevers kunnskap i de ulike fagene de har. Systemet tenkes brukt både av skolene og av elevene selv. Fagpersoner som underviser i IB-systemet skal kunne legge inn spørsmål knyttet til de fagene de selv underviser. Disse fagpersonene skal være registrert i databasen med epostadresse (som skal fungere som brukerens unike ID), passord, fullt navn, navn på institusjon han/hun er tilknyttet og hvilke fag de underviser.

Fag som inngår i studieplanen til IB, skal registreres i databasen med navn og et fagnummer. Databasen skal også holde oversikt over hvilket årstrinn (1. eller 2. år) faget gjelder for.

Fagpersonene skal kunne legge inn fire forskjellige typer spørsmål: Flervalgsspørsmål der fagpersonene kan definere to eller flere svaralternativ og der bare ett svar er riktig, flervalgsspørsmål der det kan være mer enn ett riktig svar, spørsmål der svaret skal gis som en enkelt verdi – i form av et tall, et begrep, en kjemisk formel eller liknende – og åpne prosaspørsmål der elevene skal skrive svaret som et stykke prosatekst. Fagpersonene skal også oppgi hvor mange poeng som maksimalt kan oppnås dersom oppgaven er korrekt besvart. Fagpersonene skal oppgi de(t) korrekte svar for alle spørsmål bortsett fra de åpne prosaspørsmålene. I tillegg skal spørsmålene ha en unik ID og skal klassifiseres som enkle, medium eller avanserte.

Fagpersoner kan også definere skoleprøver i et eller flere av sine fag. I ei skoleprøve inngår spørsmål som er derfinert for faget. Fagpersonene kan fritt velge om de vil benytte egne spørsmål eller spørsmål stilt av andre fagpersoner i skoleprøvene. Hver skoleprøve skal ha en tittel og en dato når den er planlagt gjennomført.

Elever i IB-systemet skal være registrert i databasen med epostadresse (som unik ID), passord, navn på videregående skole han/hun går på og årstrinn (1. eller 2. år). Databasen skal også holde oversikt over alle ganger en student har besvart et gitt spørsmål – enten som en del av en skoleprøve eller som en del av en frivillig øving, hva studenten svarte, hvilken dato spørsmålet ble besvart og hvilken score studenten oppnådde den enkelte gangen. Blanke svar skal også registreres.

Lag en konseptuell UML-modell for databasen som skal utvikles.

**b) (5%)**

Anta at leverandøren av et kalendersystem for høyskoler ønsker å bruke en database for å lagre opplysninger om hvilke forelesninger som foregår i hvilke klasserom for hver undervisningstime (nummerert fra 1 til 8 for timene som starter henholdsvis kl. 8.00 og kl. 15.00) for hver enkelt ukedag (nummerert fra 1 til 5 for dagene mandag til fredag), hvem som er faglærer og hvilke(n) klasse(r) som undervises. Anta at leverandøren foreslår følgende logiske design:

```
Tidsluke(ukeNr, dagNr, timeNr, klasse, rom, emneKode, faglærer)
PK(ukeNr, dagNr, timeNr, klasse)
```

Eksempler på tupler i relasjonen Tidsluke kan være:

```
(44, 2, 1, '10HBSPA', 'C007', 'IMT2571', 'runehj')
(44, 2, 1, '11HBDRA', 'C007', 'IMT2571', 'runehj')
(44, 2, 1, '11HBMTA', 'B110', 'IMT2261', 'haralddl')
(44, 2, 1, '12HÅRGISA', 'B110', 'IMT2261', 'haralddl')
(44, 3, 1, '10HBSPA', 'K102', 'IMT2571', 'runehj')
(44, 3, 1, '11HBDRA', 'K102', 'IMT2571', 'runehj')
(44, 3, 1, '11HBMTA', 'B110', 'IMT2261', 'haralddl')
(44, 3, 1, '12HÅRGISA', 'B110', 'IMT2261', 'haralddl')
(44, 4, 3, '11HMMTA', 'A012', 'IMT5271', 'runehj')
```

Begrepet funksjonell avhengighet (eng. *functional dependency*) er sentralt i normalisering. Forklar hva begrepet betyr.

Anta at disse funksjonelle avhengighetene skal gjelde for kalendersystemet:

```
FD1: ukeNr, dagNr, timeNr, rom → emneKode, faglærer
FD2: ukeNr, dagNr, timeNr, emneKode → rom, faglærer
FD3: ukeNr, dagNr, timeNr, faglærer → rom, emneKode
FD4: emneKode → faglærer
```

Forklar med egne ord hva hver av disse funksjonelle avhengighetene forteller oss om undervisningen som timeplanen skal designes for.

**c) (10%)**

Er relasjonen i oppgave 2b) i 3NF? Begrunn svaret. Vis hvordan relasjonen kan normaliseres til 3NF dersom den ikke er i 3NF og vis hva tuplene i de nye relasjonene vil være for eksemplene ovenfor.

**Oppgave 3 – XML (25%)**

Gitt denne DTD-en:

```
<!ELEMENT Produktkatalog (Produsenter, Produkter)>
<!ELEMENT Produsenter (Produsent*) >
<!ELEMENT Produsent (Navn) >
<!ATTLIST Produsent id ID #REQUIRED >

<!ELEMENT Produkter (ProduktGruppe | Produkt)* >
<!ELEMENT ProduktGruppe (Navn, Produkt+) >
<!ELEMENT Produkt (Kode, Navn, Beskrivelse?) >
<!ATTLIST Produkt produsentId IDREF #REQUIRED >

<!ELEMENT Navn (#PCDATA) >
<!ELEMENT Epost (#PCDATA) >
<!ELEMENT Kode (#PCDATA) >
<!ELEMENT Beskrivelse (#PCDATA) >
<!ATTLIST Beskrivelse language CDATA #IMPLIED >
```

Et valid dokument kan se ut som vist på neste side.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE Produktkatalog SYSTEM "Produktkatalog.dtd">
<Produktkatalog>
 <Produsenter>
 <Produsent id="Kingston">
 <Navn>Kingston Technology</Navn>
 </Produsent>
 </Produsenter>
 <Produkter>
 <ProduktGruppe>
 <Navn>Minnekort</Navn>
 <Produkt produsentId="Kingston">
 <Kode>605210</Kode>
 <Navn>Kingston microSDHC Card 4GB, Class4</Navn>
 <Beskrivelse language="no">
 Driftstemperatur -25-85 C
 </Beskrivelse>
 </Produkt>
 <Produkt produsentId="Kingston">
 <Kode>362738</Kode>
 <Navn>Kingston microSDHC Card 8GB, Class4</Navn>
 <Beskrivelse language="no">
 Driftstemperatur -25-85 C
 </Beskrivelse>
 </Produkt>
 <Produkt produsentId="Kingston">
 <Kode>605211</Kode>
 <Navn>Kingston microSDHC Card 16GB, Class4</Navn>
 <Beskrivelse language="no">
 Driftstemperatur -25-85 C
 </Beskrivelse>
 </Produkt>
 </ProduktGruppe>
 </Produkter>
</Produktkatalog>

```

#### a) (5%)

Skriv XPath-uttrykk som returnerer følgende informasjon fra denne typen dokumenter (se vedlegg A for en liste over funksjoner i XPath 1.0):

1. Alle produkter i produktkataloger som har norske beskrivelser (dvs. der attributtet "language" har verdien "no").
2. Navn på produktgrupper som består av mer enn fem produkter.
3. Alle produkter i produktgruppa "Minnekort" av klasse 4 eller høyere (dvs. der produktets navn ender med Class4, Class5, Class6 eller liknende).

#### b) (20%)

Anta at en bedrift jevnlig mottar ei XML-fil av typen Produktkatalog som beskrevet i starten av oppgaven, fra sine leverandører. Anta videre at bedriften har en database med data om innkjøpte produkter. I denne databasen finnes en tabell, kalt `Produkter` med kolonnene `produsent` `VARCHAR(50)`, `kode` `INT`, `navn` `VARCHAR(100)` og `beskrivelse` `VARCHAR(250)`.

Primærnøkkelen består av kolonnene `produsent` og `kode`. For eksempelet ovenfor ville tabellen se slik ut:

#### Produkt

| <u>produsent</u> | <u>kode</u> | <u>navn</u>                          | <u>beskrivelse</u>        |
|------------------|-------------|--------------------------------------|---------------------------|
| Kingston         | 605210      | Kingston microSDHC Card 4GB, Class4  | Driftstemperatur -25-85 C |
| Kingston         | 362738      | Kingston microSDHC Card 8GB, Class4  | Driftstemperatur -25-85 C |
| Kingston         | 605211      | Kingston microSDHC Card 16GB, Class4 | Driftstemperatur -25-85 C |

Innkjøpssjefen i denne bedriften ønsker å få oversikt over eventuelle nye produkter som leverandørene har i sine kataloger. Hun ønsker derfor at du skriver et PHP-skript som fjerner `Produkt`-elementer (inkludert alle barn) fra XML-dokumentet for produkter bedriften allerede har i sin database. Skriv PHP-kode for funksjonen `fjernKjenteProdukter($doc, $dbh)` der `$doc` er dokumentet åpnet i DOM-form og der `$dbh` er et PDO-objekt som er klar til bruk for å hente data fra bedriftens database. (Se vedlegg B for en liste over PDO-funksjoner og vedlegg C for en liste over PHP DOM-funksjoner.)

**Oppgave 4 – Andre databasetemaer (10%)****a) (5%)**

Hva menes med begrepet NOSQL? Beskriv noen typer NOSQL-databaser og beskriv hvorfor man i disse tilfellene gjerne ikke velger en SQL-database.

**b) (5%)**

Hva er et view i en relasjonsdatabase og hvordan kan view-mekanismen utnyttes til sikkerhetsformål?

## Vedlegg A – Oversikt over funksjoner i XPath 1.0

### Node Set Functions:

```
number last()
number position()
number count(node-set)
node-set id(object)
string local-name(node-set?)
string namespace-uri(node-set?)
string name(node-set?)
```

### String Functions:

```
string string(object?)
string concat(string, string, string*)
boolean starts-with(string, string)
boolean contains(string, string)
string substring-before(string, string)
string substring-after(string, string)
string substring(string, number, number?)
number string-length(string?)
string normalize-space(string?)
string translate(string, string, string)
```

### Boolean Functions:

```
boolean boolean(object)
boolean not(object)
boolean true()
boolean false()
boolean lang(string)
```

### Number Functions:

```
number number(object?)
number sum(node-set)
number floor(number)
number ceiling(number)
number round(number)
```

## Vedlegg B – Utdrag av PHP DOM referanse

```
DOMDocument extends DOMNode {
/* Properties */
readonly public string $actualEncoding ;
readonly public DOMConfiguration $config ;
readonly public DOMDocumentType $doctype ;
readonly public DOMELEMENT $documentElement ;
public string $documentURI ;
public string $encoding ;
public bool $formatOutput ;
readonly public DOMImplementation $implementation ;
public bool $preserveWhiteSpace = true ;
public bool $recover ;
public bool $resolveExternals ;
public bool $standalone ;
public bool $strictErrorChecking = true ;
public bool $substituteEntities ;
public bool $validateOnParse = false ;
public string $version ;
readonly public string $xmlEncoding ;
public bool $xmlStandalone ;
public string $xmlVersion ;

/* Methods */
__construct ([string $version [, string $encoding]])
DOMAttr createAttribute (string $name)
DOMAttr createAttributeNS (string $namespaceURI , string $qualifiedName)
DOMCDATASection createCDATASection (string $data)
DOMComment createComment (string $data)
DOMDocumentFragment createDocumentFragment (void)
DOMELEMENT createElement (string $name [, string $value])
DOMELEMENT createElementNS (string $namespaceURI , string $qualifiedName
 [, string $value])
DOMEntityReference createEntityReference (string $name)
DOMProcessingInstruction createProcessingInstruction (string $target
 [, string $data])

DOMText createTextNode (string $content)
DOMELEMENT getElementById (string $elementId)
DOMNodeList getElementsByTagName (string $name)
DOMNodeList getElementsByTagNameNS (string $namespaceURI ,
 string $localName)
DOMNode importNode (DOMNode $importedNode [, bool $deep])
mixed load (string $filename [, int $options = 0])
bool loadHTML (string $source)
bool loadHTMLFile (string $filename)
mixed loadXML (string $source [, int $options = 0])
void normalizeDocument (void)
bool registerNodeClass (string $baseclass , string $extendedclass)
bool relaxNGValidate (string $filename)
bool relaxNGValidateSource (string $source)
int save (string $filename [, int $options])
string saveHTML ([DOMNode $node = NULL])
int saveHTMLFile (string $filename)
string saveXML ([DOMNode $node [, int $options]])
bool schemaValidate (string $filename)
bool schemaValidateSource (string $source)
bool validate (void)
int xinclude ([int $options])
}
```



```

DOMNode {
/* Properties */
public readonly string $nodeName ;
public string $nodeValue ;
public readonly int $nodeType ;
public readonly DOMNode $parentNode ;
public readonly DOMNodeList $childNodes ;
public readonly DOMNode $firstChild ;
public readonly DOMNode $lastChild ;
public readonly DOMNode $previousSibling ;
public readonly DOMNode $nextSibling ;
public readonly DOMNamedNodeMap $attributes ;
public readonly DOMDocument $ownerDocument ;
public readonly string $namespaceURI ;
public string $prefix ;
public readonly string $localName ;
public readonly string $baseURI ;
public readonly string $textContent ;

/* Methods */
public DOMNode appendChild (DOMNode $newnode)
public string C14N ([bool $exclusive [, bool $with_comments
 [, array $xpath [, array $ns_prefixes]]]])
public int C14NFile (string $uri [, bool $exclusive
 [, bool $with_comments [, array $xpath
 [, array $ns_prefixes]]]])
public DOMNode cloneNode ([bool $deep])
public int getLineNo (void)
public string getNodePath (void)
public bool hasAttributes (void)
public bool hasChildNodes (void)
public DOMNode insertBefore (DOMNode $newnode [, DOMNode $refnode])
public bool isDefaultNamespace (string $namespaceURI)
public bool isSameNode (DOMNode $node)
public bool isSupported (string $feature , string $version)
public string lookupNamespaceURI (string $prefix)
public string lookupPrefix (string $namespaceURI)
public void normalize (void)
public DOMNode removeChild (DOMNode $oldnode)
public DOMNode replaceChild (DOMNode $newnode , DOMNode $oldnode)
}

DOMXPath {
/* Properties */
public DOMDocument $document ;

/* Methods */
public __construct (DOMDocument $doc)
public mixed evaluate (string $expression [, DOMNode $contextnode
 [, bool $registerNodeNS = true]])
public DOMNodeList query (string $expression [, DOMNode $contextnode
 [, bool $registerNodeNS = true]])
public bool registerNamespace (string $prefix , string $namespaceURI)
public void registerPhpFunctions ([mixed $restrict])
}

```

## Vedlegg C – Utdrag av PDO referanse

PDO:

```
__construct (string $dsn [, string $username [, string $password
 [, array $driver_options]]])
bool beginTransaction (void)
bool commit (void)
mixed errorCode (void)
array errorInfo (void)
int exec (string $statement)
mixed getAttribute (int $attribute)
static array getAvailableDrivers (void)
bool inTransaction (void)
string lastInsertId ([string $name = NULL])
PDOStatement prepare (string $statement
 [, array $driver_options = array()])
PDOStatement query (string $statement)
string quote (string $string [, int $parameter_type = PDO::PARAM_STR])
bool rollBack (void)
bool setAttribute (int $attribute , mixed $value)
```

PDOStatement:

```
bool bindColumn (mixed $column , mixed &$param [, int $type [, int $maxlen
 [, mixed $driverdata]]])
bool bindParam (mixed $parameter , mixed &$variable
 [, int $data_type = PDO::PARAM_STR [, int $length
 [, mixed $driver_options]]])
bool bindValue (mixed $parameter , mixed $value
 [, int $data_type = PDO::PARAM_STR])
bool closeCursor (void)
int columnCount (void)
bool debugDumpParams (void)
string errorCode (void)
array errorInfo (void)
bool execute ([array $input_parameters])
mixed fetch ([int $fetch_style
 [, int $cursor_orientation = PDO::FETCH_ORI_NEXT
 [, int $cursor_offset = 0]]])
array fetchAll ([int $fetch_style [, mixed $fetch_argument
 [, array $ctor_args = array()]]])
string fetchColumn ([int $column_number = 0])
mixed fetchObject ([string $class_name = "stdClass"
 [, array $ctor_args]])
mixed getAttribute (int $attribute)
array getColumnMeta (int $column)
bool nextRowset (void)
int rowCount (void)
bool setAttribute (int $attribute , mixed $value)
bool setFetchMode (int $mode)
```