

Databases (INFODB)

27 juni 2006

N.B. Bij dit tentamen is het toegestaan om indexcards met aantekeningen te raadplegen, die aan het eind ingeleverd dienen te worden.

Gebruikte afkortingen:

- 2PL = two-phase locking
- 3NF = derde normaalvorm
- BCNF = Boyce-Codd normaalvorm
- DP = dependency preserving
- FD = functional dependency
- RA = relationele algebra
- SQL = SQL (Structured Query Language)

Opgave 1. Algemeen

(20 punten)

- a) Bij de onderstaande vragen zijn er 0, 1 of meer antwoorden correct. Je dient voor elk antwoord aan te geven of het antwoord correct is of niet. Een toelichting is niet noodzakelijk.

Vraag 1

Welke van de onderstaande uitspraken zijn waar?

- Een relatieschema met twee attributen is altijd in 3NF.
- Elk relatieschema in BCNF is automatisch ook in 4NF.
- Een attribuut kan tegelijkertijd foreign key en primary key zijn.
- De closure van een superkey van een relatieschema is altijd de gehele attribuutverzameling.
- De closure van een candidate key van een relatieschema is altijd de gehele attribuutverzameling.

Vraag 2

Wat zijn candidate keys voor schema $R(ABCDEFG)$ met $F_R = \{A \rightarrow BC, B \rightarrow AG, CD \rightarrow FG, E \rightarrow G\}$?

- AD
- AE
- ADE
- BD

Vraag 3

We beschouwen de propagatie naar verandering van de base table(s) van een `insert`-operatie op een view. Is deze propagatie eenduidig voor achtereenvolgens een select-view, project-view en join-view?

- a) Eenduidig voor elk van de genoemde views.
 - b) Eenduidig voor de select-view, niet eenduidig voor de project-view en de join-view.
 - c) Eenduidig voor geen enkele van de genoemde views.
 - d) Eenduidig voor de select-view en de project-view, niet eenduidig voor de join-view.
 - e) Niet eenduidig voor de select-view en de project-view, eenduidig voor de join-view.
- f) Gegeven de onderstaande XML Query en de onderstaande inhoud van `http://bstore1.example.com/bib.xml`.

Wat is het resultaat in XML van deze query?

XML Query:

```
<results>
  {
    for $b in doc("http://bstore1.example.com/bib.xml")/bib/book,
      $t in $b/title,
      $a in $b/author
    return
      <result>
        { $t }
        { $a }
        <year>
          {$b/@year}
        </year>
      </result>
  }
</results>
```

`http://bstore1.example.com/bib.xml`:

```
<bib>
  <book year="1992">
    <title>Advanced Programming in the Unix environment</title>
    <author><last>Stevens</last><first>W.</first></author>
    <publisher>Addison-Wesley</publisher>
    <price>65.95</price>
  </book>
  <book year="2000">
    <title>Data on the Web</title>
    <author><last>Abiteboul</last><first>Serge</first></author>
    <author><last>Buneman</last><first>Peter</first></author>
    <author><last>Suciu</last><first>Dan</first></author>
    <publisher>Morgan Kaufmann Publishers</publisher>
    <price>39.95</price>
  </book>
</bib>
```

Opgave 2. Queries

(25 punten)

Een bedrijf dat gegevens bijhoudt over 1-daagse wielrenklassiekers heeft een database met daarin een tabel waarin gegevens van renners worden bijgehouden, een tabel waarin gegevens van wedstrijden worden bijgehouden en een tabel waarin de deelname van renners aan wedstrijden wordt bijgehouden. Elke renner heeft een uniek rennersnummer. Daarnaast wordt van elke renner de naam, het geslacht en het gewicht bijgehouden. Van elke wedstrijd wordt de naam, de datum en de lengte van het parcours bijgehouden. Eveneens heeft elke wedstrijd een uniek wedstrijdnummer. De registratie van de deelname van een renner aan een wedstrijd gaat vergezeld van de tijd waarin de renner de wedstrijd gereden heeft, de finish positie (1 voor de winnaar) en het aantal punten dat de renner in de wedstrijd behaald heeft. Indien al wel bekend is dat een renner aan een wedstrijd zal deelnemen, maar de wedstrijd nog niet gereden is, hebben deze laatste drie attributen de waarde NULL.

R (*rnr*, naam, geslacht, gewicht)

WR (*wnr*, *rnr*, tijd, positie, punten)

W (*wnr*, naam, datum, lengte)

We hebben de volgende queries:

Q1: Geef de namen van de renners die aan tenminste één wedstrijd hebben deelgenomen.

Q2: Geef de namen van de vrouwelijke winnaars die zwaarder zijn dan de zwaarste mannelijke winnaars.

Q3: Geef de namen van de renners die aan alle wedstrijden hebben deelgenomen.

Q4: Geef de namen van de renners die aan geen enkele wedstrijd hebben deelgenomen.

Q5: Geef de namen van de renners en het aantal gewonnen wedstrijden voor de renners die het grootste aantal wedstrijden gewonnen hebben.

Hieronder volgen expressies in de RA of in SQL. Geef voor elke query aan welke expressie(s) met de query corresponderen. De relatie tussen queries en expressies is many-to-many en optioneel.

E1: $\pi_{naam}((\pi_{rnr}(R) - \pi_{rnr}(WR)) \bowtie R)$

E2: $\pi_{naam}((\pi_{rnr}(R) \cap \pi_{rnr}(WR)) \bowtie R)$

E3: $\pi_{naam}((R \bowtie WR) \div \pi_{wnr}(WR))$

E4: $\pi_{naam}((\pi_{rnr}(R) \cup \pi_{rnr}(WR)) \bowtie R)$

E5: $\pi_{naam}((\pi_{rnr,wnr}(WR) \div \pi_{wnr}(WR)) \bowtie R)$

E6: SELECT R.naam
FROM R
WHERE R.rnr NOT IN
(SELECT WR.rnr
FROM WR
WHERE WR.rnr <> R.rnr)

E7: SELECT R.naam
FROM R
WHERE NOT EXISTS
(SELECT WR.rnr
FROM WR)

E8: SELECT R.naam, COUNT(WR.wnr)
FROM R,WR
WHERE R.rnr = WR.rnr
AND WR.positie = 1
GROUP BY WR.rnr
HAVING MAX(COUNT(WR.wnr))

```
E9: SELECT R.naam, COUNT(WR.wnr)
      FROM R,WR
      WHERE R.rnr = WR.rnr
      AND WR.positie = 1
      GROUP BY WR.rnr
      HAVING COUNT(WR.wnr) > ALL
             (SELECT COUNT(WR.wnr)
              FROM WR
              GROUP BY WR.rnr)
```

```
E10: SELECT R.naam
      FROM R
      WHERE R.rnr IN
             (SELECT WR.rnr
              FROM WR)
```

```
E11: SELECT R.naam, COUNT(WR.wnr)
      FROM R,WR
      WHERE R.rnr = WR.rnr
      AND WR.positie = 1
      GROUP BY WR.rnr
      HAVING COUNT(WR.wnr) >= ALL
             (SELECT COUNT(WR.wnr)
              FROM WR
              WHERE WR.positie = 1
              GROUP BY WR.rnr)
```

```
E12: SELECT R.naam
      FROM R
      WHERE NOT EXISTS
             (SELECT W.wnr
              FROM W
              WHERE NOT EXISTS
                     (SELECT WR.rnr
                      FROM WR
                      WHERE WR.rnr = R.rnr
                      AND WR.wnr = W.wnr))
```

```
E13: SELECT R.naam
      FROM R
      WHERE NOT EXISTS
             (SELECT W.wnr
              FROM W
              WHERE NOT EXISTS
                     (SELECT WR.rnr
                      FROM WR
                      WHERE WR.rnr = R.rnr))
```

```

E14: SELECT R.naam
      FROM R
      WHERE EXISTS
        (SELECT WR.rnr
         FROM WR
         WHERE WR.rnr = R.rnr)

E15: SELECT R.naam
      FROM R, WR
      WHERE R.rnr = W.rnr
      AND WR.positie = 1
      AND R.geslacht = 'Vrouw'
      AND R.gewicht >= ALL
        (SELECT R.gewicht
         FROM R, WR
         WHERE R.rnr = WR.rnr
         AND WR.positie = 1
         AND R.geslacht = 'Man')

E16: SELECT R.naam
      FROM R
      WHERE NOT EXISTS
        (SELECT WR.rnr
         FROM WR
         WHERE WR.rnr = R.rnr)

E17: SELECT R.naam, COUNT(WR.wnr)
      FROM R, WR
      WHERE R.rnr = W.rnr
      AND WR.positie = 1
      GROUP BY WR.rnr
      HAVING COUNT(WR.wnr) <= ANY
        (SELECT COUNT(WR.wnr)
         FROM WR
         WHERE WR.positie > 1
         GROUP BY WR.rnr)

```

Opgave 3. Normaalvormen

(30 punten)

We hebben een schema $R(ABCDEFG)$ en een set FD's

$F = \{BC \rightarrow DE, DE \rightarrow F, D \rightarrow FG, AC \rightarrow GE, A \rightarrow BD, G \rightarrow C\}$.

- Geef een verliesvrije, DP 3NF-decompositie van R . Laat zien welke methode je gebruikt.
- Is het resultaat van a) in BCNF? Licht toe. Zo nee, breng dit schema dan alsnog in BCNF. Laat zien welke methode je gebruikt. Is het nieuwe schema nog steeds DP?

Opgave 4. Concurrency

(25 punten)

We beschouwen de volgende twee schedules.

- Geef aan of deze schedules serialiseerbaar zijn of niet. Licht toe. Geef zo mogelijk de equivalente seriële schedules.
- Geef eveneens aan of de schedules geaccepteerd worden door een 2PL-scheduler. Geef hierbij een korte toelichting.

$S1$

T1	T2	T3	T4	T5
w(x)				
			w(y)	
	w(z)			
	w(y)			
		w(y)		
r(z)				
				r(y)

$S2$

T1	T2	T3	T4	T5
			w(x)	
w(y)				
	r(y)			
	w(z)			
				w(y)
			r(z)	
		r(y)		
r(x)				