

Tentamen Databases —6 maart 2000

De gecorrigeerde tentamens zijn af te halen op 10-04-00 bij het Onderwijsbureau, kamer 36 IWI, in de (rode) map Databases.

Opmerkingen:

- Schrijf **netjes** en duidelijk, met zwarte of blauwe pen.
- Zet op het eerste blad alle gegevens als naam, etc., en het totaal aantal ingeleverde bladen, en nummer de ingeleverde bladen.
- LET GOED OP WELKE OPGAVEN U NIET HOEFT TE MAKEN!
- Lees de opgaven eerst goed door.
- Motiveer uw antwoorden.

1. (Alleen voor Informatici)

- a) Leg uit wat de *3-schema architecture* inhoudt.
- b) Wat is het verschil tussen een B -boom en een B^+ -boom?
- c) Leg uit hoe je in een concurrent DBMS deadlock kunt voorkomen.

2. (Alleen voor Informatici)

Gegeven is een relatieschema R met attributen A, B, C, D, E . R heeft als (minimale) sleutel $\{B, C\}$.

Tussen deze attributen gelden verder de volgende (permanente) functionele afhankelijkheden (FA's): $D \rightarrow E$, $E \rightarrow AB$, $B \rightarrow C$.

- a) Waarom is dit schema niet in BCNF?
- b) Geef een minimal cover voor de FA's.
- c) Geef een decompositie in 3NF, die dependency preserving is en de nonadditive-join eigenschap bezit.

3. (Alleen voor Informatici)

Beschouw het volgende eenvoudige **netwerkmodel**:

SCHEMA NAME IS Reisburo

```
RECORD NAME IS Passagier
  LOCATION MODE IS CALC USING Pnr
    DUPLICATES ARE NOT ALLOWED FOR Pnr
    Pnr      IS TYPE Integer
    Naam     IS TYPE String(20)
    Gironr   IS TYPE String(7)
    Betaald  IS TYPE Boolean
```

```
RECORD NAME IS Vlucht
  LOCATION MODE IS CALC USING Vnr
    DUPLICATES ARE NOT ALLOWED FOR Vnr
    Vnr      IS TYPE Integer
    Van      IS TYPE String(20)
    Naar     IS TYPE String(20)
    Vertrek  IS TYPE Date
    Aankomst IS TYPE Date
```

```
RECORD NAME IS Piloot
  LOCATION MODE IS CALC USING Naam
    DUPLICATES ARE NOT ALLOWED FOR Naam
    Naam     IS TYPE String(20)
    Brevet   IS TYPE String(10)
```

```
SET NAME IS Boeking
  OWNER IS Vlucht
  MEMBER IS Passagier
    INSERTION IS MANUAL
    RETENTION IS OPTIONAL
```

```
SET NAME IS Vliegt
  OWNER IS Piloot
  MEMBER IS Vlucht
    INSERTION IS MANUAL
    RETENTION IS OPTIONAL
```

Maak een applicatie, die de volgende query beantwoordt:

“Geef de namen van alle piloten, die vandaag (Date=06032000) naar New York vliegen, die Brevet=“Boeing-747” hebben.”

(Hint: U dient dus gebruik te maken van commando's als FIND FIRST, FIND NEXT, FIND OWNER, GET etc. etc.)

Voor een bedrijf met vrij veel afdelingen, verdeeld over een gering aantal managers, als beleidslijn, definiëren we een database-universum UMA over het database-skelet GMA:

$$\begin{aligned}
 \text{GMA} = \{ & \\
 (\text{MW} & : \{\text{MNR}, \text{MNAAM}, \text{SAL}, \text{AFDNR}\}), \\
 (\text{AFD} & : \{\text{ANR}, \text{ANAAM}, \text{BUDGET}, \text{MNR}\}) \\
 & \}
 \end{aligned}$$

$$\begin{aligned}
 \text{FMW} = \{ & \\
 (\text{MNR} & : \text{Vng}(5)), & & : \text{medewerker-nummer} \\
 (\text{MNAAM} & : \text{Chs}(30)), & & : \text{naam} \\
 (\text{SAL} & : N), & & : \text{salaris} \\
 (\text{AFDNR} & : \text{Vng}(3)) & & : \text{afdeling} \\
 & \}
 \end{aligned}$$

$$\begin{aligned}
 \text{FAFD} = \{ & \\
 (\text{ANR} & : \text{Vng}(3)), & & : \text{afdeling-nummer} \\
 (\text{ANAAM} & : \text{Chs}(30)), & & : \text{naam} \\
 (\text{BUDGET} & : N), & & : \text{salaris-budget} \\
 (\text{MNR} & : \text{Vng}(5)) & & : \text{manager} \\
 & \}
 \end{aligned}$$

$$\begin{aligned}
 \text{TMW} & = \prod(\text{FMW}) \\
 \text{TAFD} & = \prod(\text{FAFD})
 \end{aligned}$$

$$\begin{aligned}
 \text{WMW} & = \{T \subseteq \text{TMW} \mid \{\text{MNR}\} \text{ is u.i. in } T\} \\
 \text{WAFD} & = \{T \subseteq \text{TAFD} \mid \{\text{ANR}\} \text{ is u.i. in } T\}
 \end{aligned}$$

$$\begin{aligned}
 \text{HMA} = \{ & \\
 (\text{MW} & : \text{WMW}), & & : \text{medewerkers} \\
 (\text{AFD} & : \text{WAFD}) & & : \text{afdelingen} \\
 & \}
 \end{aligned}$$

$$\begin{aligned}
 \text{UMA} = \{v \mid v \in \prod(\text{HMA}) \text{ en} & \\
 \{(\text{MNR}:\text{MNR})\} \text{ verbindt } v(\text{AFD}) \text{ met } v(\text{MW}), \text{ en} & \quad (1) \\
 \{(\text{AFDNR}:\text{ANR})\} \text{ verbindt } v(\text{MW}) \text{ met } v(\text{AFD}) & \quad (2) \\
 & \}
 \end{aligned}$$

4. (Alleen voor TBW'ers)

Geef, uitgaande van het database-universum UMA, zoals hierboven gedefinieerd, de volgende verzameling eenduidig in woorden weer:

$$\left\{ \begin{array}{l} \cup(G \parallel \{MNR, MNAAM, SAL\}) \\ \cup\{(AA: | G |), (TB: \sum t \in G : t(BUDGET))\} \\ | G \in Part(v(AFD) \bowtie v(MW), \{MNR\}) \text{ en } | G | \geq 3 \\ \} \end{array} \right.$$

5. (Zowel voor TBW'ers als Informatici)

Geef van de formele weergave van de query in opgave 4 een zo rechtstreeks mogelijke "doorvertaling" naar SQL(2).

6. (Zowel voor TBW'ers als Informatici)

Geef de hieronder gevraagde query formeel weer in termen van het database-universum UMA:

"Geef nummer en naam van elke afdelingsmanager van wie bij geen van zijn of haar afdelingen het salarisbudget is overschreden."

7. (Alleen voor TBW'ers)

Definieer een database-universum UMA2, die ten opzichte van UMA aan de volgende extra constraint voldoet (voor zover UMA daaraan nog niet voldeed): "Iedereen die manager van een afdeling is, moet in ieder geval ook manager zijn van de afdeling waaraan hij of zij formeel verbonden is."