

Hertentamen Databases —16 mei 2003

De gecorrigeerde tentamens zijn af te halen bij het Onderwijsbureau, kamer 36 IWI, in de (rode) map Databases, resp. bij het Buro Onderwijs en Examens van TBW, WSN 640.

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!
- Motiveer uw antwoorden.

1. (Alleen voor Informatici)

Gegeven is het tabel-skelet $R = \{A, B, C, D, E\}$, en de verzameling functionele afhankelijkheden $F = \{AB \rightarrow CD, B \rightarrow DE, C \rightarrow E\}$.

- a). Geef een minimale sleutel voor R .
- b). Geef een minimal cover voor F .
- c). Geef een lossless-join dependency preserving decompositie van R in 3NF.

2. (Alleen voor Informatici)

Wat is dynamic hashing? Leg uit hoe deze methode werkt. Je mag je beperken tot invoegen van nieuwe elementen.

3. (Alleen voor Informatici)

Beschouw het volgende **netwerk**model:

```
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)
```

```
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 (netwerk-query), die het volgende beantwoordt:
"Geef de namen van alle piloten die een vliegtuig vliegen met een
passagier met de naam Smit aan boord."

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

We definiëren een database-universum UMA over het database-skelet GMA ter registratie van de medewerkersinzet bij projecten van de afdelingen van een organisatie:

```
GMA = {
(MW ; {MNR, MNAAM, SAL, AUW, ANR } ),
(AFD ; {ANR, ANAAM, BUDGET, MANNR } ),
(PR ; {PNR, PNAAM, VANR } ),
(PM ; {PNR, MNR, UPW, ROL } )
}
```

```
FMW = {
(MNR ; Snn(5)), ; medewerkernummer
(MNAAM; Chs(30)), ; naam
(SAL ; N), ; maandsalaris in euro's
(AUW ; [1. .40]), ; contractueel aantal diensturen per week
(ANR ; Snn(3)) ; afdeling
}
```

```
FAFD = {
(ANR ; Snn(3)), ; afdelingnummer
(ANAAM ; Chs(30)), ; naam
(BUDGET; N), ; maandsalaris-budget in euro's
(MANNR ; Snn(5)) ; manager
}
```

```
FPR = {
(PNR ; Snn(4)), ; projectnummer
(PNAAM; Chs(30)), ; naam
(VANR ; Snn(3)) ; verantwoordelijke afdeling
}
```

```
FPM = {
(PNR ; Snn(4)), ; project
(MNR ; Snn(5)), ; medewerker
(UPW ; [1. .40]), ; uren inzet per week
(ROL ; Chs(20)) ; medewerkersrol in project
}
```

$$\begin{aligned}
WMW &= \{T \subseteq \prod(FMW) \mid \{MNR\} \text{ is u.i. in } T\} \\
WAFD &= \{T \subseteq \prod(FAFD) \mid \{ANR\} \text{ is u.i. in } T\} \\
WPR &= \{T \subseteq \prod(FPR) \mid \{PNR\} \text{ is u.i. in } T\} \\
WPM &= \{T \subseteq \prod(FPM) \mid \{PNR, MNR, ROL\} \text{ is u.i. in } T \text{ en} \\
&\quad \{PNR\} \text{ is u.i. in } \{t \in T \mid t(ROL) = \text{'PL'}\}\}
\end{aligned}$$

$$\begin{aligned}
HMA &= \{ \\
& (MW ; WMW), && \text{; medewerkers} \\
& (AFD ; WAFD), && \text{; afdelingen} \\
& (PR ; WPR), && \text{; projecten} \\
& (PM ; WPM), && \text{; projektdeelnames} \\
& \}
\end{aligned}$$

$$\begin{aligned}
UMA &= \{v \mid v \in \prod(HMA) \text{ en} \\
& \quad \{(MANNR;MNR)\} \text{ verbindt } v(AFD) \text{ met } v(MW), \text{ en} && (1) \\
& \quad \{(ANR;ANR)\} \text{ verbindt } v(MW) \text{ met } v(AFD), \text{ en} && (2) \\
& \quad \{(VANR;ANR)\} \text{ verbindt } v(PR) \text{ met } v(AFD), \text{ en} && (3) \\
& \quad \{(PNR;PNR)\} \text{ verbindt } v(PM) \text{ met } v(PR), \text{ en} && (4) \\
& \quad \{(MNR;MNR)\} \text{ verbindt } v(PM) \text{ met } v(MW) && (5) \\
& \}
\end{aligned}$$

Toelichting: De rol 'PL' is (het enige) synoniem voor projectleider.

4. (Alleen voor TBW'ers)

Geef, uitgaande van het database universum UMA, zoals hierboven gedefinieerd, de volgende verzameling duidelijk en eenduidig in gewoon Nederlands weer:

$$\begin{aligned}
& \lambda v \in UMA : \\
& \{ (p \uparrow \{PNR, PNAAM\} \cup a \uparrow \{ANR, ANAAM\} \\
& \quad \mid p \in v(PR) \text{ en } a \in v(AFD) \text{ en } p(VANR) = a(ANR) \text{ en} \\
& \quad \mid \{d \mid d \in v(PM) \bowtie v(MW) \text{ en } d(PNR) = p(PNR) \\
& \quad \quad \text{en } d(ANR) \neq p(VANR)\} \mid = 0 \\
& \}
\end{aligned}$$

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 van elke afdeling met een maandsalarisbudget van meer dan € 100.000 het afdelingsnummer, de afdelingsnaam, het maandsalarisbudget en de totale som van de feitelijke maandsalarissen van alle medewerkers van die afdeling.”

7. (Alleen voor TBW'ers)

We willen een aantal constraints toevoegen aan UMA. Geef voor elk van die constraints aan hoe en waar UMA aangepast moet worden (tenzij UMA daaraan al voldeed):

- (a) Een medewerker mag binnen een project niet meerdere rollen vervullen.
- (b) Een medewerker kan alleen maar deelnemen aan projecten van zijn of haar eigen afdeling.
- (c) Elk project heeft een projectleider.