

# Data & Informatie voorbeeld van Toets 2

## Antwoorden

### Databaseontwerp en SQL

Bij het tentamen mogen geen boeken, aantekeningen of elektronische apparaten worden gebruikt.

#### Antwoord 1

Beoordeling: 16 punten voor de FDs (2 punt aftrek voor ieder fout antwoord W/O en voor iedere foute motivatie); 4 punt voor de MVDs.

Vraag	FD/MVD	W/O	Motivatie
a)	$V \rightarrow TAP$	O	regel b; ook: een vestiging kan meerdere producten hebben (3 en 4)
b)	$TAP \rightarrow V$	W	uit regel a volgt $T \rightarrow V$ ; en daaruit volgt $TAP \rightarrow V$ .
c)	$P \rightarrow M$	O	bij een product (dus één of misschien wel meer afdelingen, zie 4) kunnen meerdere medewerkers horen, zie 5.
d)	$M \rightarrow P$	O	bij een medewerker (dus afdeling, volgens e) kunnen meerdere producten horen volgens 4.
e)	$E \rightarrow T$	W	Volgens regels d, e, 3, a geldt: $E \rightarrow M$ , $M \rightarrow A$ , $A \rightarrow V$ , $V \rightarrow T$ .
f)	$T \rightarrow E$	O	bij een tel. nummer (dus uniek bepaalde vestiging, zie 2 en a) kunnen meerdere medewerkers horen (zie 5) dus ook emails (zie 6).
g)	$V \twoheadrightarrow T$	W	Uit a en 2 volgt $V \rightarrow T$ . Elke FD is ook een MVD
h)	$A \twoheadrightarrow VTP$	W	Voor een enkele afdeling A zijn de medewerkers (en hun emailadressen) onafhankelijk van de producten P. A is geen key van AVTP (bij een afdeling horen meerdere producten). Er geldt wel $A \rightarrow V$ , $V \rightarrow T$ en dus $A \rightarrow T$ . De join AVTP $\bowtie$ AME levert dus geen tuples op die niet in de oorspronkelijke tabel zitten. A is ook geen key van AME (bij een afdeling horen meerdere medewerkers).
i)	$E \twoheadrightarrow M$	W	Volgens d: $E \rightarrow M$ . Elke FD is ook een MVD.

#### Antwoord 2

**Antwoord a)** ABD en ACD zijn de enige sleutels. (Attributen A en D komen niet in het rechterlid van een FD voor en behoren dus tot iedere sleutel, en net zo behoort B of C tot iedere sleutel.) Van de gegeven  $\mathcal{F}$  zijn alle leden een schending van de BCNF-eis voor R; bijvoorbeeld voor  $AB \rightarrow CE$ : de FD is niet-triviaal en het linkerlid omvat geen sleutel (dat wil zeggen, het linkerlid is geen supersleutel).

**Antwoord b)** We passen een stap van het BCNF-algoritme toe. We kiezen (zomaar) de eerste schending,  $AB \rightarrow CE$ , ter eliminatie. Dus delen we R op in  $R_1 = ABCE$  (alle attributen van de schending), en  $R_2 = ABDF$  (het linker lid van de schending, plus alle overgebleven attributen in R). Voor de functionele afhankelijkheden  $\mathcal{F}_1$  en  $\mathcal{F}_2$  behorende bij  $R_1$  en  $R_2$  geldt:  $\mathcal{F}_1 = \{AB \rightarrow CE\}$  en

$\mathcal{F}_2 = \{AB \rightarrow F\}$ . Let op: de FD  $AB \rightarrow F$  zit weliswaar niet in  $\mathcal{F}$ , maar de FD zit wel in de closure  $\mathcal{F}^+$  en dus, omdat al zijn attributen in  $R_2$  zitten, ook in  $\mathcal{F}_2$ .

(Een mogelijk andere decompositie wordt verkregen door een andere schending te elimineren. Zie het alternatieve antwoord bij het laatste onderdeel.)

**Antwoord c)** De afhankelijkheid  $CD \rightarrow BF$  is niet behouden:  $CD \rightarrow BF \notin (\mathcal{F}_1 \cup \mathcal{F}_2)^+$ .

De afhankelijkheden  $E \rightarrow F$  en  $F \rightarrow E$  zijn niet behouden:  $E \rightarrow F \notin (\mathcal{F}_1 \cup \mathcal{F}_2)^+$  en  $F \rightarrow E \notin (\mathcal{F}_1 \cup \mathcal{F}_2)^+$ .

**Antwoord d)** We passen het BCNF-algoritme toe.

- De eerste stap is in de vorige deelvraag gedaan en levert de volgende decompositie:  
 $R_1 = ABCE$ ;  $\mathcal{F}_1 = \{AB \rightarrow CE\}$  en  $R_2 = ABDF$ ;  $\mathcal{F}_2 = \{AB \rightarrow F\}$ .
- In  $R_1$  is  $AB$  een sleutel, dus is  $AB \rightarrow CE$  geen schending van de BCNF-conditie, en dus staat  $R_1$  in BCNF.
- In  $R_2$  is  $AB \rightarrow F$  een schending van de BCNF-conditie, want  $AB \rightarrow F$  is niet-triviaal en  $AB$  is geen supersleutel in  $R_2$  (want  $AB^+ = ABF \neq ABDF$ ). We kiezen  $AB \rightarrow F$  ter eliminatie.
- Dus delen we  $R_2$  op in  $R_{2a} = ABF$  (alle attributen van de schending) en  $R_{2b} = ABD$  (het linker lid van de schending, plus alle overgebleven attributen in  $R_2$ ). Voor de functionele afhankelijkheden  $\mathcal{F}_{2a}$  en  $\mathcal{F}_{2b}$  behorende bij  $R_{2a}$  en  $R_{2b}$  geldt:  $\mathcal{F}_{2a} = \{AB \rightarrow F\}$  en  $\mathcal{F}_{2b} = \{ \}$ .
- Dus  $R_1$ ,  $R_{2a}$  en  $R_{2b}$  vormen een lossless decompositie van  $R$ . Er geldt:  $R = R_1 \bowtie R_{2a} \bowtie R_{2b}$ .  
Ofwel:  $ABCDEF = ABCE \bowtie ABF \bowtie ABD$ .

Wanneer je met een andere schending begint kun je mogelijk een andere decompositie bereiken, en een andere uitslag voor het behoud van de functionele afhankelijkheden.

**Alternatief antwoord** Een andere decompositie van  $R$  wordt verkregen door de schending  $E \rightarrow F$  als uitgangspunt te kiezen. We krijgen dan  $R_1 = EF$ ,  $\mathcal{F}_1 = \{E \rightarrow F, F \rightarrow E\}$  en  $R_2 = ABCDE$ ,  $\mathcal{F}_2 = \{AB \rightarrow CE, CD \rightarrow BE\}$ . In dit geval blijven alle FDs behouden (ook  $CD \rightarrow F$ ). Van deze schema's staat  $R_1$  in BCNF en  $R_2$  niet. Verdere decompositie van  $R_2$  aan de hand van  $AB \rightarrow CE$  levert  $R_{2a} = ABCE$ ,  $\mathcal{F}_{2a} = \{AB \rightarrow CE\}$ , en  $R_{2b} = ABD$ ,  $\mathcal{F}_{2b} = \{ \}$ . Beide staan in BCNF, en  $CD \rightarrow B$  en  $CD \rightarrow F$  zijn niet behouden.

## Antwoord 3

**Antwoord a)** (15 punten)

```
SELECT s.name FROM Ship s, Outcome o, Battle b
WHERE s.name = o.shipname
AND b.name = o.battlename
AND b.date = '1700-04-01'
AND o.result = 'sunk';
```

**Antwoord b)** (15 punten)

```
SELECT c.country, COUNT(DISTINCT o.battlename)
FROM Class c, Ship s, Outcome o
WHERE c.name = s.classname AND s.name = o.shipname
GROUP BY c.country
HAVING MIN(s.launched) < 1600
```

**Antwoord c)** (20 punten)

Shunting rule tweemaal (and if perfect commutativity of 'AND')

```
SELECT DISTINCT b.name FROM Battle b, Ship s1, Ship s2, Outcome o1, Outcome o2
WHERE o1.shipname = s1.name
AND o2.shipname = s2.name
AND o1.battlename = b.name
AND o2.battlename = b.name
AND s1.classname <> s2.classname;
```

o1.battlename = b.name:

```
SELECT DISTINCT o1.battlename FROM Battle b, Ship s1, Ship s2, Outcome o1, Outcome o2
WHERE o1.shipname = s1.name
AND o2.shipname = s2.name
AND o1.battlename = b.name
AND o2.battlename = o1.battlename
AND s1.classname <> s2.classname;
```

Shunting Battle back (and commutativity of AND):

```
SELECT DISTINCT o1.battlename FROM Ship s1, Ship s2, Outcome o1, Outcome o2
WHERE EXISTS (
  SELECT * FROM Battle b
  WHERE o1.battlename = b.name
)
AND o1.shipname = s1.name
AND o2.shipname = s2.name
AND o2.battlename = o1.battlename
AND s1.classname <> s2.classname;
```

Foreign key rule FOREIGN KEY (o1.battlename) REFERENCES Battle(name):

```
SELECT DISTINCT o1.battlename FROM Ship s1, Ship s2, Outcome o1, Outcome o2
WHERE o1.shipname = s1.name
AND o2.shipname = s2.name
AND o2.battlename = o1.battlename
AND s1.classname <> s2.classname;
```

**Vraag d)** (bonusvraag: 5 extra punten – maakt het mogelijk om meer dan 100 punten te halen)

```
SELECT c0.c FROM Class c0
WHERE NOT EXISTS (
  SELECT * FROM Ship s, Class c
  WHERE s.c = c.s AND c.c = c0.c
  AND NOT EXISTS (
    SELECT * FROM Battle b
    WHERE b.date < 1600
    AND s.n = o.s
    AND o.b = b.n
    AND o.r = damaged
  )
);
```