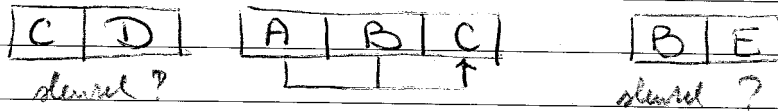
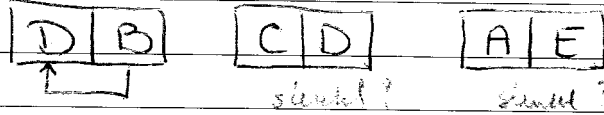


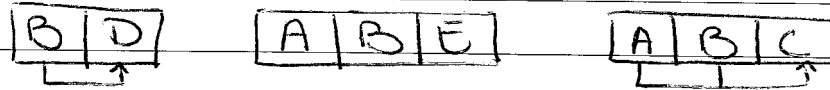
1



in BCNF, want AB is superkey in {A,B,C}



in BCNF, want B is superkey in {D,B}



ook in BCNF, want B is superkey in {B,D} en A,B superkey in {A,B,C}

2 dus a, b en c zitten in BCNF ^{niet waar!}
ik kies antwoord d, want ik mag niet a, b en c kiezen!

2 BCNF is strikter dan 3NF, dus a is niet waar
 4NF is strikter dan BCNF, dus b is niet waar
 maar dus wel 4NF \Rightarrow BCNF,
 dus ook 3NF \wedge 4NF \Rightarrow BCNF
 dus c is waar \int

3 d is niet waar! \int

tegenvoorbeeld: $R = \{A, B, C\}$ } niet in BCNF
 $F = \{AB \rightarrow C, C \rightarrow B\}$ }

wanneer je een ~~table~~ tabel maakt met niet alle attributen, gaat er minstens een functionele afhankelijkheid verloren

c is niet waar! alles in een dergelijke tabel is afhankelijk van een superkey en dat is niet noodzakelijkerwijs een primaire sleutel ook \int

vb: $R = \{A, B, C\}$
 $F = \{A \rightarrow BC, AB \rightarrow C\}$ met A primaire sleutel
 $(AB)^+ = R$

alles is dus ook afhankelijk van AB en niet alleen van A

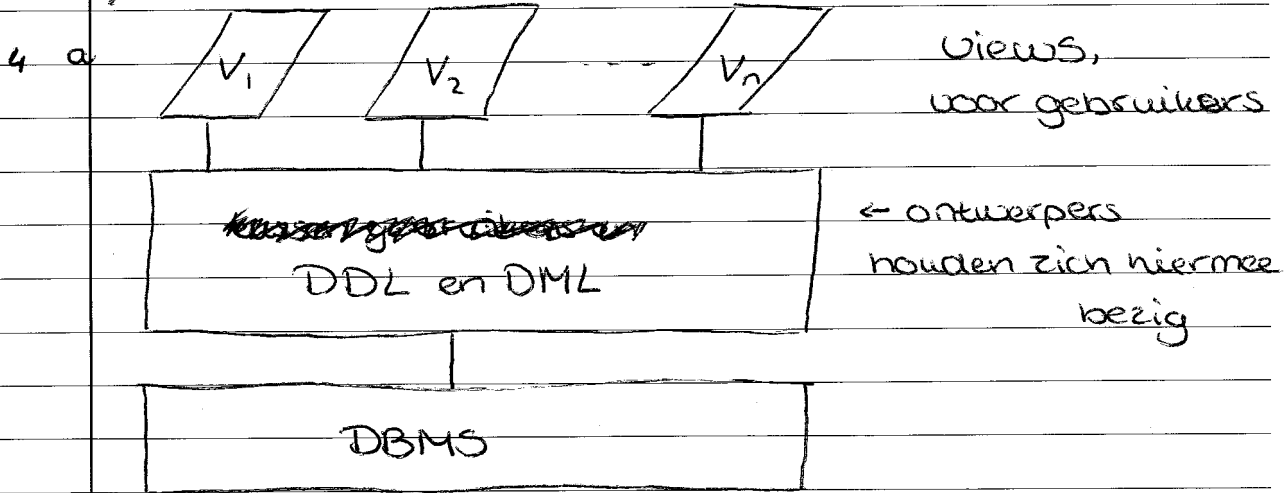
vervolg 3

b: lossless join kan in 3NF

lossless join kan in BCNF \Rightarrow 3NF

duis b is ^{mogelijk} waar, maar ik kan er niet veel _{mit} meer over zeggen

a is waar



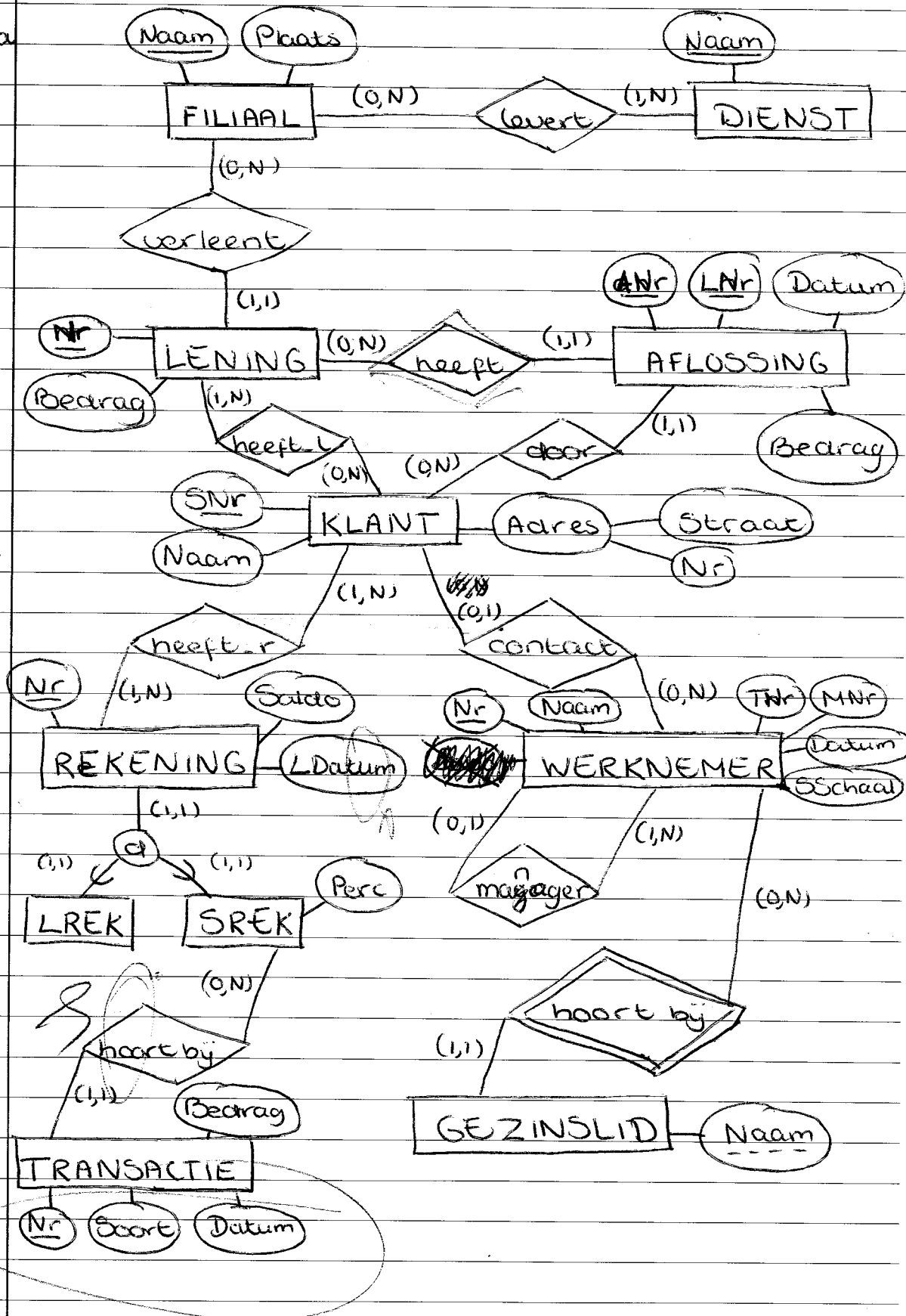
b soms moeten dingen op verschillende niveaus aangepast worden, het is wenselijk dat men er op een ander niveau geen last van heeft en dat de database door kan blijven draaien

verder moeten de verschillende lagen natuurlijk goed op elkaar aansluiten

c - verschillende views voor gebruikersgroepen

gebruikers ^{hoeven} ~~kennen~~ dus ^{over} niet ~~van~~ onderliggende lagen te weten

5 a



8

~~1~~

10

7

b DIENST Naam

Revert Filiaal ~~Naam~~ Dienst

FILIAAL Naam Plaats

Verleent Filiaal Lening

LENING Nr Bedrag

heeft - l Klant Lening

KLANT SNr Naam Straat Nr

doet Klant Aflossing Lening

AFLOSSING ANr LNr Datum Bedrag Klant

heeft r Klant Rekening

REKENING Nr Saldo Datum

SREK Nr Perc

LREK Nr

~~Mede~~ TRANSACTIE Nr Bedrag Soort Datum ~~Rnr~~

contact Klant Werknemer

WERKNEMER SNr Naam TNr MNr Datum SSchaal

Hoofd bij Werknemer
GEZINSLID Werknemer Naam

geen normale relatie?

Afdeling Wiskunde en Informatica R.U.G.

| | | |
|-------------|-------------------------------|--------------|
| Naam: | Studentnummer: | Bladnr.: 3 |
| Adres: | Studierichting: | Tentamen: |
| Postcode en | Jaar van eerste inschrijving: | Datum: |
| Woonplaats: | | Naam docent: |

6

```

Vlucht. Van := 'Schiphol';
$ FIND ANY Vlucht USING Van;
while DB-STATUS = 0 do
begin
$ GET Vlucht;
$ FIND FIRST WITHIN Boeking USING Van Naam;
Passagier. Naam := 'Jansen';
if DB-STATUS = 0 then
begin
$ GET OWNER WITHIN vliegt;
$ FIND GET Pilot;
writeln(Piloot. Naam);
end;
$ FIND DUPLICATE Vlucht USING Van;
end;

```

10 a, b en c doen SELECT <tabel>, dat voldoet niet aan de SQL-syntax, dus antwoord d

12 d, want koppelingen en tests zijn nergens gebaseerd op maker, dat is hier wel nodig, nl. : $7(\text{model1} \leftrightarrow \text{model2} \Rightarrow \text{maker}_1 \langle \rangle \text{maker}_2)$

15 ~~#~~ c is niet waar, want ongelijke modellen zegt niets over de fabrikant

b is dus ook niet waar

d is het goede antwoord! a
(bij a wordt namelijk ook niet gekeken naar de fabrikant)

19 c niet goed, zegt niets over maker

b slaat helemaal nergens op!
(Product in WHERE-clause maar niet in FROM)

a is goed! ~~8~~
(aantal PC-makers > 1 mag niet, daar zijn er 0 van)