

[Essential Skills] Mathematics

*Studiewijzer*

Module [3]

*Predicatenlogica*

Tijd: 420 minuten

## ▼ 3.0 Leerdoelen en onderwerpen

### ▼ 3.0.1 Leerdoelen

Na deze module bestudeerd te hebben moet je:

- kunnen definiëren wat een predicaat is en hoe een predicaat overgaat in een propositie;
- inzien hoe predicaten kunnen worden gebruikt bij het aantonen van de correctheid van algoritmen;
- variabelen in een predicaat kunnen binden door toekennen van een waarde of door kwantificatie;
- de universele en existentiële kwantor kennen en kunnen toepassen;
- het verschil tussen vrije en gebonden (dummy) variabelen kennen en deze in een expressie kunnen herkennen;
- het belang inzien van de volgorde waarin kwantoren in een predicaat worden genoteerd;
- natuurlijke taal en kwantoren in elkaar kunnen overzetten;
- combinaties van kwantoren met logische operatoren kunnen evalueren;
- het begrip scope van een kwantor kunnen toepassen.

### ▼ 3.0.2 Onderwerpen

3.1 <i>Binding van variabelen .....</i>	3
3.2 <i>Toekennen van een waarde .....</i>	5
3.3 <i>Binding met behulp van een kwantor .....</i>	8
3.4 <i>Predicaat met verschillende variabelen .....</i>	13
3.5 <i>Volgorde van kwantoren .....</i>	15
3.6 <i>Bewijzen en weerleggen .....</i>	17
3.7 <i>Antwoorden van de opgaven .....</i>	18

### 3.1 Binding van variabelen

In sommige situaties kunnen beweringen nodig zijn zoals:  $a + b = 7$ ,  $x^2 > 10$  of 'Zij is lief'. Deze beweringen zijn *geen* proposities omdat ze niet true of false zijn. Als we echter een waarde toekennen aan de variabelen uit deze uitdrukkingen worden het proposities. Dit soort expressies noemen we *predicaten*.

Let op dat variabelen in de wiskunde niet hetzelfde zijn als variabelen in een programmeertaal. Dit uit zich o.a. in het feit dat aan variabelen in een programmeertaal steeds andere waarden kunnen worden toegekend (b.v.  $n := 7$  of  $n := n + 1$ ), terwijl variabelen in de wiskunde (vaak) staan voor een hele verzameling: 'laat  $n$  een natuurlijk getal zijn, dan...'.

**Definitie.** Een predicaat is een expressie met variabelen, die een propositie wordt als we waarden aan de variabelen toekennen.

**Programmeervoorbeeld.** Vooral in een programma spelen predicaten een grote rol. In de opdracht

```
IF (x<3) AND (z=2)
THEN x:=(x+z)
ELSE x:=(x+7)
```

zijn ' $x < 3$ ' en ' $z = 2$ ' predicaten, die tijdens de uitvoering van het programma proposities worden (en dus een waarde true of false krijgen) omdat de variabelen  $x$  en  $z$  dan aan een waarde worden gebonden.

Predicaten kunnen ook worden gebruikt om de toestand van de variabelen die een rol spelen in een programma te beschrijven. Stel dat we de som van de waarden uit een invoerrij moeten bepalen. In zo'n geval kan het volgende predicaat helpen om zinvol over de correctheid van een programma te praten:

'  $S$  is de som van de eerste  $n$  ingelezen waarden '.

Het is gebruikelijk om een predicaat af te korten met een hoofdletter. Zo kan ' $a + b = 7$ ' worden genoteerd als  $S(a, b)$ ,  $x^2 > 10$  als  $P(x)$  en 'Ze is lief':  $L(x)$ .

In het algemeen worden predicaten genoteerd als:

$$P(x_1, x_2, \dots, x_n)$$

en noemt men  $x_i$  een variabele.  $P$  wordt dan genoemd een predicaat met  $n$  variabelen.

De waarden die aan de variabelen kunnen worden toegekend moeten komen uit een welomschreven collectie van waarden. We noemen dit het *universum* of *domein*, of de *individue-verzameling*. Zo zal het weinig zinvol zijn om in het predikaat ' $a + b = 7$ ' voor  $a$  te kiezen 'trein' en voor  $b$  'appel'.

Het toekennen van een waarde aan een variabele wordt het binden van de variabele genoemd. Pas als alle variabelen gebonden zijn, wordt een predicaat een propositie. Indien door binding van variabelen de waarde van het predikaat true wordt, zeggen we dat het predikaat geldt of dat de gekozen waarde voldoet. Indien het predikaat de waarde true heeft voor alle waarden van de variabelen uit het domein, dan is het predikaat geldig binnen het domein; als het voldoet voor sommige (niet noodzakelijk voor alle) dan is het vervulbaar en als het voor geen enkele waarde voldoet, dan is het onvervulbaar.

Het binden van variabelen kan op twee manieren gebeuren:

- door het toekennen van een waarde, of
- door het gebruik van een kwantor.

In de volgende hoofdstukken worden deze manieren toegelicht.

## 3.2 Toekennen van een waarde

Beschouw het predicaat  $P(a, b) : 'a + b = 7'$ . Indien we aan  $a$  de waarde 3 toekennen en aan  $b$  de waarde 4, dan wordt het predikaat een propositie met waarheidswaarde true. Verkort genoteerd:  $P(3, 4) \leftrightarrow \text{true}$ . Dit laatste mogen we op grond van de false-true regels ook schrijven als  $P(3, 4)$ . Verder zijn natuurlijk ook de volgende uitspraken true:  $P(2, 5)$ ,  $P(7, 0)$ ,  $\neg P(9, 2)$ ,  $\neg P(100, 7)$ , enz.

**Programmeervoorbeeld.** Stel dat  $P$  een predikaat is met twee variabelen:

$P(s, n) : 's$  is de som van de positieve getallen uit de  $n$  getallen die zijn ingelezen'.

We kunnen door binding van  $s$  en  $n$  van het predikaat een ware propositie maken (immers de som van nul getallen is 0):  $s := 0; n := 0$ .

Na deze twee toekenningsoopdrachten zal gelden:

$P(s, n) \Leftrightarrow \text{true}$ .

We kunnen deze constructie verder gebruiken bij de ontwikkeling van een (stukje van een) programma. In de bij het programma opgenomen commentaarregels, is aangegeven welke toestandsveranderingen door de verschillende toekenningsoopdrachten worden veroorzaakt (we noemen dit *annotatie*).

```
{true}
s:=0;
n:=0;
{P(s,n)}
read(a) ;
n:=n+1;
{P(s,n-1)}
IF a>0 THEN s:=s+a;
{P(s,n)}
```

Het is op dit moment goed om in te zien, dat hier een toekenning van  $n+1$  aan  $n$  niet wordt beschreven door te zeggen 'de waarde van  $n$  wordt met 1 verhoogd' (een dynamische beschrijving), maar dat er een beschrijving wordt gegeven van de toestand van de variabelen (door middel van een predicaat) voor en na de toekenningsoopdracht (een statische beschrijving).

*Boolean-notaties*

Hier alvast een waarschuwing, die verwijst naar het vak programmeren. De directe vertaling van het begrip logische propositie in Java of in bijv. C++ is de zogenaamde

boolean-variabele. Het is een standaard type in Java, dat slechts twee waarden aan kan nemen: true en false. In de programmeervoorbeelden is tot nu toe al verschillende keren de algemene structuur (als bijvoorbeeld in PASCAL)

```
if <voorwaarde>
then <actie1>
else <actie2>
```

gebruikt. Deze structuur luidt in Java en C++ :

```
if ( <voorwaarde> )
    <statement_1>;
else
    <statement_2>;
```

Op de plaats van <voorwaarde> staat een boolean-variabele, of een uitdrukking die een boolean-variabele oplevert. Bv.:

```
if ( doorgaan ) of if ( x>3 )
    ... ;          .... ;
else
    ... ;          else
    .... ;
```

Als doorgaan de waarde true aanneemt wordt de eerste statement uitgevoerd, anders die na else. De notatie  $P(3, 4) \Leftrightarrow true$  geeft misschien aanleiding tot een directe vertaling in een programmeertaal. Maar het is onzin om de <voorwaarde> te vertalen als:

```
if ( doorgaan == true)
```

Immers doorgaan is een boolean-variabele die zelf al true of false is. En de uitdrukking:

```
doorgaan == true
```

levert precies dezelfde boolean-waarde op als 'doorgaan' alléén. Om dezelfde reden is:

```
if ((x>3) == true)
```

een gedrocht, dat er nog rotter uitziet. Om het helemaal duidelijk te maken: je schrijft toch ook niet:

```
if ((doorgaan = true) = true)
```

In de programmeerlessen zul je ongetwijfeld voor dit soort constructies worden teruggefloten. In de wiskunde kun je dus wèl  $P(3, 4) \Leftrightarrow \text{true}$  gebruiken, om aan te geven dat de waarheidswaarde van de propositie  $P(3, 4)$  true is.



### 3.3 Binding met behulp van een kwantor

De tweede methode die wordt gebruikt om variabelen te binden is de kwantificatie van de variabelen. Het predikaat ' $x^2 > 0$ ' is bijvoorbeeld geldig voor (reële) waarden van  $x$  en het predikaat ' $x + y \geq 100$ ' voor waarden van  $x$  die ten minste gelijk zijn aan  $(100 - y)$ , enz. Ook kan het gewenst zijn de uitspraak te doen dat een predikaat niet voor *alle* waarden true is, maar ook niet voor alle waarden false. Denk hierbij aan het predikaat ' $x$  is een kwadraat' of ' $x$  is alleen deelbaar door 1 en  $x$ '.

#### Universele kwantor

De binding voor alle waarden kan worden gedaan met behulp van de universele kwantor of 'Allemaal', aangeduid met  $\forall$ . Stel dat we een predikaat hebben beschreven als  $P(i)$ . Indien we nu een uitspraak willen doen over alle waarden van  $i$  uit een bepaald domein (aangegeven met  $d(i)$ ), dan noteren we dat als:

$$(\forall i : d(i) : P(i))$$

uit te spreken als:

*Voor alle waarden van  $i$  die voldoen aan de domeinbeschrijving  $d(i)$ , geldt  $P(i)$ .*

Deze uitspraak is nu een propositie en kan dus weer true of false zijn. We kunnen de universele kwantor zien als een verkorte notatie voor een conjunctie. Neem b.v. de uitspraak dat voor alle waarden van  $i$  uit het domein  $0 \leq i \leq 10$  geldt: ' $i^2 \geq 50$ '. Introduceer het predikaat:  $P(i) : i^2 \geq 50$ , dan noteren we:

$$P(0) \wedge P(1) \wedge P(2) \wedge \dots \wedge P(10) \leftrightarrow (\forall i : 0 \leq i \leq 10 : P(i)).$$

*Opmerking 1:* Dit is alleen geldig voor een eindig domein.

*Opmerking 2:* Merk echter wel op dat  $(\forall i : 0 \leq i \leq 10 : P(i))$  false is.

**Voorbeelden.** Enkele voorbeelden van het gebruik van de universele kwantor:

- $(\forall x : 2 \leq x \leq 10 : x^2 \geq 4)$
- $(\forall x : x > 10 : 2 \cdot x > 15)$
- $(\forall x : 0 \leq x \leq 10 : 'x \text{ is een kwadraat}')$
- $(\forall x : x \text{ is een mens} : 'x \text{ heet Jan}')$
- $(\forall a : a \text{ is een auto} : 'a \text{ rijdt net zo goed als en Volvo}')$

Van deze proposities (want het zijn nu geen predikaten meer!) zijn de eerste twee true en de laatste drie false.

*Existentiële kwantor*

Indien we willen formuleren dat een predikaat waar is voor (tenminste) een van de waarden uit het domein van de variabele, dan kunnen we gebruik maken van de existentiële kwantor ('Er is een waarde') aangeduid met  $\exists$ . Deze kwantor is te beschouwen als een verkorte notatie voor de disjunctie van een aantal proposities.

We definiëren de uitspraak:  $P(a) : 'a \text{ is een priemgetal}'$ .

*Ter herinnering:*

Een priemgetal is een (positief) getal dat precies twee verschillende delers heeft.

De priemgetallen zijn dus, voor zover nog niet duidelijk: 2, 3, 5, 7, 11, 13, 17, 19, .....

Er is dus slechts één even priemgetal (2) en het getal 1 is geen priemgetal. (1 heeft immers niet twee verschillende delers).

Er geldt dan (bijvoorbeeld):  $P(5) \vee P(6) \vee P(7) \vee P(8) \vee P(9)$  immers 5 en 7 zijn priemgetallen; dus  $P(5) \Leftrightarrow \text{true}$  en  $P(7) \Leftrightarrow \text{true}$ .

Met behulp van een kwantor noteren we:  $(\exists a : 5 \leq a \leq 9 : P(a))$ .

In 'gewoon Nederlands' betekent dit: *Er is een priemgetal met een waarde ten minste 5 en ten hoogste 9.*

Algemene notatie:

$$(\exists i : d(i) : P(i))$$

uit te spreken als:

*Er is een waarde voor  $i$  die voldoet aan de domeinbeschrijving  $d(i)$  waarvoor geldt  $P(i)$ .*

*Opmerking:* We beschouwen ook hier slechts eindige domeinen.

**Voorbeelden.**

$(\exists x : 0 \leq x \leq 10 : 'x \text{ is een kwadraat}')$

$(\exists x : x \geq 10 : x = 12)$

$(\exists x : x < 0 : \neg (x < -10))$

$(\exists x : 0 \leq x \leq 10 : x \text{ div } 10 = 2)$

$(\exists x : x > 10 : x + 1)$

$(\exists a : 'a \text{ is een land in Europa}' : 'In a \text{ wonen geen mensen}')$

Ook hier zijn de eerste drie proposities true en de laatste drie false.

*Intermezzo*

In het voorbeeld en de extra voorbeelden hieronder worden de uitdrukkingen div en mod gebruikt. Deze uitdrukkingen worden in dit intermezzo nader verklaard. Ze hebben veel te maken met wat op de basisschool geleerd wordt met delen. In dat geval is er sprake van een

uitkomst (quotiënt) en een rest. In eerste instantie kun je de *div*-uitdrukking gebruiken voor de uitkomst. Zo geldt dat  $7 \text{ div } 2 = 3$ . De *mod*-uitdrukking definieert de rest. Dus  $7 \text{ mod } 2 = 1$ .

De volledige definities:

**Definitie.** De uitdrukking  $n \text{ div } k$  is het grootste gehele getal, ten hoogste gelijk aan  $\frac{n}{k}$ , waarbij  $n \geq 0$  en  $k > 0$ .

**Definitie.** De uitdrukking  $n \text{ mod } k$  is het kleinst mogelijke gehele getal, ten minste 0, dat overblijft als  $n$  met een  $k$ -voud wordt verminderd of vermeerderd ( $k > 0$ ).

De uitdrukking *div* wordt ook wel eens genoemd gehele deling (of heling) en *mod* de rest na gehele deling.

### Extra voorbeelden.

$$21 \text{ mod } 4 = 1$$

$$21 \text{ div } 4 = 5$$

$$(55 \text{ div } 4) \text{ mod } 2 = 1$$

$$'x \text{ is even}' \Leftrightarrow 'x \text{ mod } 2 = 0'$$

### Unieke kwantor

Soms wordt nog gebruik gemaakt van de unieke kwantor die een uitspraak doet over het feit dat er precies een waarde uit het domein is, waarvoor het predikaat voldoet. Deze kwantor wordt geschreven als  $\exists!$  en uitgesproken als *Er is precies een waarde uit het domein beschreven door  $d(i)$ , waarvoor geldt  $P(i)$ .*

### Voorbeeld.

$$(\exists! i : i > 10 : i^2 = 625)$$

### Dummy variabele

Door de binding van een variabele met behulp van een kwantor, is deze variabele buiten de kwantor 'niet meer bekend'. Dat betekent dat we voor die variabele een willekeurige vorm kunnen kiezen, zonder dat daardoor de waarheidswaarde van de propositie verandert.

$$(\forall i : d(i) : P(i))$$

is ook te schrijven als:

$$(\forall x : d(x) : P(x)).$$

De variabelen  $i$  en  $x$  in deze proposities zijn door binding *dummy* of *gebonden* variabelen geworden. Indien in een predikaat nog een variabele voorkomt die niet door de kwantor

wordt gebonden, noemen we die variabele een *vrije* variabele. Hoewel de keuze van de representatie van de dummy variabele binnen een kwantor vrij is, moeten we toch wel goed uitkijken:

$$(i > 10) \wedge (\forall i : d(i) : P(i))$$

is een toegestane uitdrukking en er wordt mee bedoeld dat een of andere variabele  $i$  groter dan 10 is en dat het predikaat  $P$  voor alle waarden uit het domein van de propositievariabele true is (er vanuit gaande dat de gehele uitdrukking true is). De  $i$  uit de eerste operand heeft niets te maken met de  $i$  uit de tweede operand. In zo'n situatie is het verstandiger deze conjunctie te noteren als:

$$(i > 10) \wedge (\forall k : d(k) : P(k)).$$

### ▼ **Opgave 3.3.1**

Gegeven de volgende predikaten:

$B(x) : x$  is een boek

$N(x) : x$  is geschreven in het Nederlands

$E(x) : x$  is geschreven in het Engels

$D(x) : x$  is dik

Vertaal de volgende logische expressies in goed Nederlands:

- a)  $(\forall x : B(x) : E(x))$
- b)  $(\exists x : B(x) : D(x) \wedge E(x))$
- c)  $(\forall x : B(x) : D(x) \rightarrow E(x))$
- d)  $\neg (\exists x : B(x) : D(x) \wedge E(x))$
- e)  $(\forall x : B(x) : N(x) \vee E(x))$

[Maak op Maple T.A. de oefening: \[3.3.1\] Logische expressies naar zinnen vertalen](#)

### ▼ *Opgave 3.3.2*

Schrijf de volgende Nederlandse zinnen met behulp van kwantoren. Gebruik de predikaten van opgave 3.3.1.

- a) *Er zijn alleen Nederlandse boeken*
- b) *Er is een Engels boek*
- c) *Er is geen dik Nederlands boek*
- d) *Er is geen boek dat zowel in het Nederlands, als in het Engels is geschreven*
- e) *Alle Engelse boeken zijn dik*

[Maak op Maple T.A. de oefening: \[3.3.2\] Zinnen naar logische expressies vertalen](#)

### 3.4 Predicaat met verschillende variabelen

Indien in een predicaat verschillende variabelen voorkomen, moet men goed in de gaten houden welke variabelen door een kwantor worden gebonden (de *dummy* variabelen) en welke de *vrije* variabelen zijn.

Stel dat we als predicaat gebruiken:

$$P(x, y) : 'x + y > 0'$$

dan gaat dit predicaat pas over in een propositie indien *beide* variabelen op de een of andere manier worden gebonden.

**Voorbeeld.** Gebruik voor  $P(x, y) : 'x + y > 0'$ .  $P(2, 3)$  is een propositie;  $P(-10, 2)$  ook;  $P(x, 7)$  is geen propositie omdat  $x$  niet is gebonden.  $(\exists x : x > 0 : P(x, 7))$  is wel een propositie omdat  $y$  is gebonden door het toekennen van de waarde 7, terwijl  $x$  is gebonden door een kwantor.

**Voorbeeld.** Gebruik voor  $P(x, y, z) : 'x + y = z'$ .

$P(x, y, z)$ ;  $x, y$  en  $z$  zijn alle drie vrije variabelen.

$P(2, y, z)$ ;  $y$  en  $z$  zijn vrije variabelen.

$(\exists y : y > 0 : P(x, y, z))$ ;  $y$  is gebonden,  $x$  en  $z$  zijn vrije variabelen.

$(\forall x : x > 0 : (\exists y : y > 0 : (\exists z : z > 0 : P(x, y, z))))$ ;  $x, y$  en  $z$  zijn alle drie door een kwantor gebonden, dus is de uitdrukking een propositie geworden.

Ga na dat de waarheidswaarde van het laatste voorbeeld true is.

De laatste vorm in het voorbeeld mag ook afgekort worden tot:

$$(\forall x \exists y \exists z : x > 0 \wedge y > 0 \wedge z > 0 : P(x, y, z)).$$

Indien we van deze notatie gebruik maken, kunnen we een eerder gegeven voorbeeld, waarin natuurlijke taal werd gebruikt, formeler weergeven:

$$(\exists i : 0 \leq i \leq 10 : 'i \text{ is een kwadraat}')$$

kan ook als volgt worden weergegeven:

$$(\exists i : 0 \leq i \leq 10 : (\exists j : j \geq 0 : i = j^2))$$

en dus ook als:

$$(\exists i \exists j : 0 \leq i \leq 10 \wedge j \geq 0 : i = j^2))$$

of zelfs als:

$$(\exists i, j : 0 \leq i \leq 10 \wedge j \geq 0 : i = j^2))$$

### ▼ 3.5 Volgorde van kwantoren

Indien verschillende kwantoren worden gebruikt, kan de volgorde waarin de kwantoren worden genoteerd een grote rol spelen.

$$(\forall x \exists y : 'x \text{ is een mens}' \wedge 'y \text{ is een vrouw}' : 'y \text{ is de moeder van } x')$$

is een bewering die luidt: *Voor iedereen geldt dat men een moeder heeft.* De waarheidswaarde van deze bewering is natuurlijk true, maar door het verwisselen van de kwantoren ontstaat een bewering die false is:

$$(\exists y \forall x : 'x \text{ is een mens}' \wedge 'y \text{ is een vrouw}' : 'y \text{ is de moeder van } x')$$

want er staat nu: *Er is een vrouw, die moeder is van alle mensen,* en dit is uiteraard niet waar.

De oorzaak van dit verschil is dat in de tweede propositie de keuze van  $y$  ( $y$  is de moeder van  $x$ ) onafhankelijk kan worden gedaan van de keuze van  $x$ , terwijl in het eerste geval de keuze van  $y$  afhangt van de keuze van  $x$ .

#### Voorbeelden.

Nog enkele voorbeelden (het domein voor de variabelen wordt gevormd door de gehele getallen en is in de voorbeelden weggelaten):

$$(\forall x \exists y :: x + y = 0) \Leftrightarrow \text{true, neem } y = -x.$$

$$(\exists y \forall x :: x + y = 0) \Leftrightarrow \text{false, er is geen } y \text{ waarvoor geldt dat de som met elk ander geheel getal 0 oplevert.}$$

$$(\exists !x \forall y :: x \cdot y = 0) \Leftrightarrow \text{true, neem } x = 0, \text{ dit is tevens de enige } x \text{ die voldoet, vandaar } \exists !$$

$$(\forall y \exists !x :: x \cdot y = 0) \Leftrightarrow \text{false, indien } y = 0, \text{ geldt het predicaat voor elke waarde van } x.$$

Deze voorbeelden illustreren het feit dat de volgorde van de kwantoren van belang is en natuurlijk ook welke variabele door welke kwantor gebonden wordt.

*Let op:* als het duidelijk is uit welk domein de variabelen komen, laten we de vermelding van dat domein dus vaak weg. Als het domein bijvoorbeeld de natuurlijke getallen is dan wordt:

$$(\forall i : i \in \mathbb{N} : i^2 \geq 0) \text{ ook } (\forall i :: i^2 \geq 0)$$

(let op de dubbele dubbele punt). We hebben dus een propositie, die een samenstelling is van oneindig veel proposities, telkens met het 'en'-teken als operator. We nemen aan dat we dit soort 'oneindige' uitspraken ook wel begrijpen!



▼ **Opgave 3.5.1**

Vertaal onderstaande proposities in goed lopende Nederlandse zinnen en geef telkens aan of de waarheidswaarde van de propositie true of false is. Leg telkens uit waarom.

- a)  $(\forall x \exists y : x, y \in \mathbb{Z} : y = 3x)$
- b)  $(\exists x \forall y : x, y \in \mathbb{Z} : y = 3x)$
- c)  $(\exists x \forall y : x, y \in \mathbb{Z} : x \cdot y = 0)$

[Maak op Maple T.A. de oefening: \[3.5.1\] Volgorde van kwantoren](#)

▼ **Opgave 3.5.2**

Onderzoek de waarheidswaarde van de volgende proposities, als het domein van  $x$ ,  $y$  en  $z$  de gehele getallen zijn:

- a)  $(\forall x \exists! y :: x + y < 0)$
- b)  $(\forall x \forall y \exists! z :: x + y = z)$
- c)  $(\forall x \exists! z \forall y :: x + y = z)$

[Maak op Maple T.A. de oefening: \[3.5.2\] Waarheidswaarde van proposities bepalen](#)

## 3.6 Bewijzen en weerleggen

Het onderzoeken van de waarheidswaarde van gekwantificeerde uitspraken zoals in bovenstaand voorbeeld kan lastig werk zijn. In het algemeen kun je het volgende daarover zeggen. Een  $\forall$ -uitspraak is vaak moeilijk te bewijzen, dat wil zeggen ervan vaststellen dat hij de waarde true heeft. Zeker als het domein van de kwantor een grote verzameling is; je moet dan immers bewijzen dat de uitspraak voor alle elementen van het domein waar is, en dat kan een oneindig werk zijn. Een  $\forall$ -uitspraak weerleggen is vaak veel eenvoudiger, dat wil zeggen ervan vaststellen dat hij de waarde false heeft. Je hoeft dan immers maar één element uit het domein aan te wijzen, waarvoor de uitspraak false is.

Andersom is het bewijzen van een  $\exists$ -uitspraak vaak eenvoudig; wijs maar één element aan dat 'het doet' en je bent klaar. Het weerleggen van zo'n  $\exists$ -uitspraak is juist vaak lastig, want je moet alle elementen controleren. Bij dit soort werk wordt natuurlijk vaak gebruik gemaakt van allerlei wiskundige kennis en technieken, om (oneindig) lange klussen toch snel te kunnen klaren.

## ▼ 3.7 Antwoorden van de opgaven

### ▼ Opgave 3.3.1

- a) Alle boeken zijn geschreven in het Engels.
- b) Er is een dik, Engels boek.
- c) Alle dikke boeken zijn geschreven in het Engels.
- d) Er is geen dik, Engels boek.
- e) Alle boeken zijn geschreven in het Nederlands of in het Engels.

### ▼ Opgave 3.3.2

- a)  $(\forall x : B(x) : N(x))$
- b)  $(\exists x : B(x) : E(x))$
- c)  $\neg (\exists x : B(x) : D(x) \wedge N(x))$
- d)  $\neg (\exists x : B(x) : N(x) \wedge E(x))$
- e)  $(\forall x : B(x) : E(x) \rightarrow D(x))$

### ▼ Opgave 3.5.1

- a) Elk getal heeft een drievoud. Dat is uiteraard waar.
- b) Er is een getal dat van elk willekeurig getal het derde deel is. Dat is niet waar. Want stel dat  $x = a$  voldeed; dan was voor alle  $y$ :  $y = 3a$ . Dat is niet waar (bijvoorbeeld niet voor  $y = 3a - 1$ )
- c) Er is een getal dat vermenigvuldigd met een willekeurig getal altijd 0 oplevert. Waar, neem maar  $x = 0$ .

### ▼ Opgave 3.5.2

Onderzoek de waarheidswaarde van de volgende proposities, als het domein van  $x$ ,  $y$  en  $z$  de gehele getallen zijn:

- a) Neem  $x = -y - 1$  en  $x = -y - 2$ , dan is het waar voor alle  $y$ , maar voor meer dan 1 waarde van  $x$ , dus niet waar.
- b) Neem  $z = x + y$ , dan is het waar.
- c) Niet waar, want er staat dat voor elke waarde van  $x$  er precies 1 waarde voor  $z$  kan worden gekozen, zodat welke waarde voor  $y$  er ook bij  $x$  wordt opgeteld het resultaat  $z$  is (neem bijvoorbeeld  $x = 0$ ).