



Samenvatting

TI1306 Redeneren & Logica

Review Guide 2014

Door: David Alderliesten

Disclaimer

De informatie in dit document is afkomstig van derden. W.I.S.V. 'Christiaan Huygens' betracht de grootst mogelijke zorgvuldigheid in de samenstelling van de informatie in dit document, maar garandeert niet dat de informatie in dit document compleet en/of accuraat is, noch aanvaardt W.I.S.V. 'Christiaan Huygens' enige aansprakelijkheid voor directe of indirecte schade welke is ontstaan door gebruikmaking van de informatie in dit document.

De informatie in dit document wordt slechts voor algemene informatie in dit documentdoeleinden aan bezoekers beschikbaar gesteld. Elk besluit om gebruik te maken van de informatie in dit document is een zelfstandig besluit van de lezer en behoort uitsluitend tot zijn eigen verantwoordelijkheid.

Review Guide – Redeneren en Logica

Redeneringen

Een redenering bestaat uit **premissen**, gevolgd door een **conclusie**. Premissen en conclusies moeten **proposities** zijn, dus moeten ze waar of onwaar zijn.

Logische Geldigheid

Logische geldigheid is een situatie waarin als alle premissen waar zijn, dat dan ook de conclusie waar is.

Het lastige van logische geldigheid is dat een redenering logisch geldig is ook als het niet zo lijkt. Het is niet belangrijk of de premissen en de conclusie ook echt waar zijn, maar alleen dat de conclusie waar is als de premisse dat zijn.

Een redenering is logisch geldig als er een **bewijs** bestaat. Een redenering is niet logisch geldig als er een **tegenvoorbeeld** bestaat.

Stelling

Een stelling is een redenering die logisch geldig is.

Bewijs en bewijstechnieken

Een bewijs laat zien dat een redenering geldig is voor alle mogelijkheden van de premissen. Om bewijzen te maken moet je kijken naar de **vormen** eromheen.

Negatie / Niet A / $\neg A$

Conjunctie / A en B / $A \wedge B$

Disjunctie / A of B / $A \vee B$

Implicatie / Als A, dan B / $A \rightarrow B$

Bi-implicatie / A, desda B / $A \leftrightarrow B$

Indirect en Directe bewijzen

Bij een direct bewijs begin je met de premissen, en redeneer je rechtstreeks naar de conclusie toe.

Bij een indirect bewijs laat je zien dat de conclusie niet onwaar kan zijn, door uit de aanname dat de conclusie onwaar is een tegenspraak af te leiden.

Tegenspraak

Een tegenspraak is wanneer een propositie en de negatie van een propositie allebei waar zijn.

Voldoende en Noodzakelijk

Een voldoende voorwaarde is een geval waarin P voldoende is om ook Q waar te laten zijn. Denk hierbij aan: "als p, dan q."

Een noodzakelijke voorwaarde is een geval waarin de contrapositie ook waar is. Denk hierbij aan: "als p, dan q... dus als q niet waar, dan p ook niet waar."

Pariteit (Parity)

Twee gehele getallen hebben dezelfde pariteit als ze **beide** even of oneven zijn.

Als X en Y gehele getallen zijn zodanig dat $X + Y$ even is, dan hebben X en Y dezelfde pariteit.

Implicaties en de “negaties” ervan

Als een implicatie “als P dan Q” waar is:

- Is de **contrapositie** “als niet Q, dan niet P” ook waar.
- Is de **converse** “als Q dan P” niet per se waar.

Als de implicatie en de converse van die implicatie waar zijn, dan zijn P en Q **equivalent**. Dit is ook bekend als een **bi-implicatie**. P is dan noodzakelijk en voldoende voor Q, en Q is noodzakelijk en voldoende voor P.

Desda

“P desda Q” betekent:

- P dan als Q
 - Dit is de implicatie als Q dan P.
- P slechts dan als Q
 - Dit is “als niet Q dan niet P” oftewel “als P dan Q.”

Disjunctie en “of” bewijzen

Als A niet waar is, dan B waar is, dan is A of B waar.

Er bestaat een X waarvoor geldt...

Het idee is dat er hier een bewijs moet komen dat een x bestaat. Dit kan op twee manieren:

- Constructief: Laat zien dat er een X bestaat.
- Niet-constructief: Laat zien dat de aanname dat zo’n X niet bestaat een tegenspraak wordt.

Voor alle X geldt

Redeneer over de **typische elementen** van C. “Neem een willekeurig element van C, zeg K.” K is hier typisch.

Notatie voor eigenschappen en predicaten

Eigenschappen en predicaten worden vaak aangeduid (als een wiskundige functie) met de volgende notatie:

Even zijn: $E(2)$, $E(3)$.

Wit zijn: $W(\text{amerigo})$, $W(\text{blackbeauty})$

Groter zijn dan: $G(2, 4)$, $G(2, 2)$

Notaties van standards

- N : natuurlijke getallen $(0, 1, 2, 3, \dots)$
- Z : gehele getallen $(\dots, -3, -2, -1, 0, 1, 2, 3, \dots)$
- Q : rationale getallen $(a \mid p, q \neq 0, aq = p)$
- R : reële getallen
- C : complexe getallen $(a + bi \mid a, i^2 = -1)$

Bewijs met inductie

- Basisstap
 - Bewijs de $P(n)$ voor $n = 0$ of $n = 1$.
- Inductiestap
 - Voor alle $n \geq 0$ geldt, als $P(n)$, dan $P(n+1)$. Neem een willekeurig getal $K \geq 0$. Stel: $P(k)$ is waar via de inductiehypothese, dus [stelling met $k+1$]. TB: $P(k+1) =$ [stelling met $k+1$].
- Eindstap
 - Omdat K willekeurig was, geldt voor alle $n \geq 0$, als $P(n)$, dan $P(n+1)$. **Volgens het inductie principe geldt $P(n)$ voor alle n element van N .**

Sommatie notatie

$$\sum_{i=1}^n i = 1 + 2 + 3 + 4 + \dots + n$$

Waarheidsfunctionele connectieven

Connectieven waarbij de waarheid van de samengestelde propositie uitsluitend afhangt van de **waarheid** van de samenstellende proposities zijn waarheid functionele connectieven. Let op, het gaat dus **niet over een verband tussen de proposities!**

Negatie / Niet A / $\neg A$

Conjunctie / A en B / $A \wedge B$

Disjunctie / A of B / $A \vee B$

Implicatie / Als A, dan B / $A \rightarrow B$

Bi-implicatie / A, desda B / $A \leftrightarrow B$

Waarheidstafel

Een waarheidstafel definieert de waarheidswaarde van de connectieven van de proposities.

P	Q	R	Formule
0	0	0	...
0	0	1	...
0	1	0	...
0	1	1	...
1	0	0	...
1	0	1	...
1	1	0	...
1	1	1	...

Implicaties zijn waar voor $0 \rightarrow 0$, $0 \rightarrow 1$, en $1 \rightarrow 1$. Bi-implicaties zijn waar voor $0 \leftrightarrow 0$, en $1 \leftrightarrow 1$.

Tautologie, Contradictie, en Contingentie

- Een propositie A is een tautologie desda A **niet onwaar** kan zijn.
- Een propositie A is een contradictie desda A **niet waar** kan zijn.
- Een propositie A is een contingentie desda A **niet een tautologie of een contradictie** is.
- Twee proposities A en B zijn logisch equivalent als $A \leftrightarrow B$ een tautologie is.
 - Wordt ook wel geschreven als $A = B$.

Logische geldigheid

Een redenering is correct of logisch geldig wanneer we gedwongen zijn de conclusie voor waar aan te nemen als we de premissen voor waar aannemen. Oftewel, logische geldigheid bestaat als voor redeneringen $A_0, A_1, \dots, A_n$ waar zijn, dat dan conclusie B ook waar is.

- Als alle premissen in een waarheidstafel de waarde "1" hebben, en de conclusie ook "1," dan is de redenering logisch geldig.
- Als alle premissen in een waarheidstafel de waarde "1" hebben, en de conclusie een "0," dan is de redenering ongeldig.
 - Dit is ook een tegenvoorbeeld.

Syntaxis van de propositielogica

Elke taal heeft een **syntaxis**, wat we **opschrijven**, en een syntax, wat we bedoelen.

Het alfabet van de propositielogica bestaat uit:

- Propositionsymbolen = $P_0, P_1, P_2, \dots$
- Connectieven = $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- Hulpymbolen = "(" en ")"

PROP

De verzameling PROP van formules van P is de kleinste verzameling rijtjes over het alfabet van P zodanig dat:

- P_i een element is van PROP, voor iedere i die een element is van N .
- Als A, B een element zijn van PROP, dan:
 - $\neg A$
 - $A \vee B$
 - $A \wedge B$
 - $A \rightarrow B$
 - $A \leftrightarrow B$
 - Een element van PROP.

Subformules

Een formule A is een subformule van F als A voorkomt in de constructie van F. Preciezer, A is een subformule van F als een van de volgende voorwaarden geldt:

- $A = F$
- $F = \neg C$, en A is een subformule van C.
- $F = (C * D)$, en A is een subformule van C en/of van D.

Recursieve definities van formules

Een recursieve definitie is een definitie die die functie **zelf aanroept/gebruikt**.

$$\begin{aligned} F(n) &= 2^n \\ F(0) &= 1 && \text{(randvoorwaarde)} \\ F(n) &= 2 * F(n-1) && \text{(Voor } N \geq 1) \end{aligned}$$

Recursieve definities van PROP

Alle formules in PROP kunnen worden uitgedrukt met de volgende methodes:

- P_i
- $\neg A$
- $(A * B)$

Structurele Inductie

Structurele inductie is een bewijstechniek om te bewijzen dat alle formules in PROP een eigenschap hebben.

- Basisstap, dus dat $P(p_i)$ geldt.
- Inductiestap, dus voor $E(A)$ en $E(B)$, $E(\neg A)$, en $E(A * B)$.
- “Volgens het principe geldt $E(F)$ voor alle F die een element zijn van PROP.”

Semantiek van P, valuaties

De waarheidswaarde die formule F krijgt is slechts afhankelijk van de **waarheidswaarden** van de **propositievariabelen** in F. Een valuatie V is een functie die aan iedere propositievariabele in p_i van P een waarheidswaarde $v(p_i) = \{0, 1\}$ toekent.

Valuaties in bewijzen als waarheidsfuncties

- $F(x) = x$
- $F_{\neg}(x) = (1 - x)$
- $F_{\wedge}(x, y) = \max(x, y)$
- $F_{\vee}(x, y) = \min(x, y)$
- $F_{\rightarrow}(x, y) = \max(1-x, y)$
- $F_{\leftrightarrow}(x, y) = \begin{cases} 1, & \text{als } x = y \\ 0, & \text{anders} \end{cases}$

Waarheidswaarde van een formule

De waarheidswaarde van een formule met betrekking tot een gegeven valuatie v kan nu recursief worden gedefinieerd als een functie. $V : \text{PROP} \rightarrow B$, $B = \{0, 1\}$:

- $V(p_i) = v(p_i)$
- $V(\neg A) = f_{\neg}(v(A))$
- $V((A * B)) = f_*(v(a), v(b))$

Model en Vervulbaarheid

- Een **model** van formule A is een **valuatie v met $v(A) = 1$** ; A is vervulbaar, v vervult A.
- Een valuatie v is een model van een verzameling formules Γ als v een model is voor alle formules in Γ .
- Een formule A is vervulbaar als A een model heeft, anders is hij **onvervulbaar**.

Tautologie, tegenmodel

- Een **tegenmodel** is een valuatie v die een model is voor alle formules van A , maar niet voor C .
- Een **tautologie**, $\models F$, is een formule waarbij elke valuatie een model is voor F .
 - Wordt soms aangeduid als **algemeen geldig**.

Metabewering

Een metabewering is een bewering die waar is voor alle proposities. Een voorbeeld:

Voor alle $A, B = \{\text{PROP}\}$ geldt: als $A \models B$, dan $\models (A \rightarrow B)$

Tegenvoorbeeld

Een tegenvoorbeeld tegen de geldigheid van een redenering bestaat uit:

- Een keuze van een formule uit een metervariabele.
- Een uitleg dat voor deze formules de uitspraken niet waar zijn.

Vervulbaarheid en onvervulbaarheid

- Als $\{A_1 \dots A_n, C\}$ vervulbaar is, zegt het niks over de geldigheid.
- Als $\{A_1 \dots A_n, C\}$ onvervulbaar is, zegt het niks over de geldigheid.
- Als $\{A_1 \dots A_n, \neg C\}$ vervulbaar is, is de redenering niet geldig.
- Als $\{A_1 \dots A_n, \neg C\}$ onvervulbaar is, is de redenering geldig.

Boommethode

De boommethode is een methode waarmee je kunt nagaan of een formule **C logisch gevolg** is van een verzameling formules uit Γ .

“ $\Gamma \models C$ desda $\Gamma \cup \{\neg C\}$ is niet vervulbaar.”

De boommethode onderzoekt of een verzameling van formules vervulbaar is, en zo niet, welke tegenvoorbeelden er bestaan. Schrijf eerste **alle premissen**, en dan de **negatie van de conclusie**. Als een tak niet sluit, is de redenering niet geldig. Als alle takken wel sluiten, is de redenering wel geldig.

Formule van regel	Uitkomst
$\neg\neg A$	A
$A \wedge B$	A B
$\neg(A \wedge B)$	$\neg A \vee \neg B$
$A \vee B$	TWEE TAKKEN: Een voor A, een voor B.
$\neg(A \vee B)$	$\neg A \wedge \neg B$
$A \rightarrow B$	$\neg A \vee B$
$\neg(A \rightarrow B)$	$A \wedge \neg B$
$A \leftrightarrow B$	$A \rightarrow B$ $B \rightarrow A$
$\neg(A \leftrightarrow B)$	$\neg(A \rightarrow B) \vee \neg(B \rightarrow A)$

Vergeet niet om bij elke regel (ook bij takken) een regelnummer te schrijven, en ook de $\neg$ -elim en regelnummers erbij te schrijven.

Natuurlijke deductie / Fitch

Fitch is een bewijssysteem dat werkt met afleidingsregels. Introductie en eliminatie regels. Het doel is om de tweede/laatste formule af te leiden uit de eerste/aanname formule. Je maakt voor elke aanname een nieuwe “draad,” en probeert de gewilde formule af te leiden.

$(A \wedge B) \models ((C \wedge D) \rightarrow (E \wedge F))$

| $A \wedge B$

|| $C \wedge D$

||| C

|||D

||E ^ F

|||E

|||F

Regels van Fitch

V-intro		V-elim	¬-intro	¬-elim
A	B	$A \vee B$	A	$\neg\neg A$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$A \vee B$	$A \vee B$	A	B	A
		$\vdots$	$\neg B$	
		C	$\vdots$	
		B	$\neg A$	
		$\vdots$		
		C		
		$\vdots$		
		C		

→-intro	→-elim	∧-intro	∧-elim	
A	$A \rightarrow B$	A	$A \wedge B$	$A \wedge B$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
B	A	B	A	B
$\vdots$	$\vdots$	$\vdots$		
$A \rightarrow B$	B	$A \wedge B$		

↔-intro	↔-elim		rei	
$A \rightarrow B$	$A \leftrightarrow B$	$A \leftrightarrow B$	A	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$B \rightarrow A$	$A \rightarrow B$	$B \rightarrow A$	$\vdots$	A
$\vdots$			$\vdots$	
$A \leftrightarrow B$			$\vdots$	

Verzamelingen

Een verzameling is een **veelheid** beschouwd als een. Simpelweg, het is een collectie van elementen en/of objecten die verzameld zijn als een. Een verzameling wordt volledig bepaald door zijn elementen.

$N = \{0, 1, 2, 3, \dots\}$ – *Natuurlijke getallen*

$B = \{0, 1\}$ – *Waarheidswaarden/Booleaans*

$Z = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$ – *alle gehele getallen*

Kenmerken van verzamelingen

- De volgorde van de elementen van de verzameling doen er niet toe/maken niks uit.
- Een object kan maar een keer in een verzameling zitten.
 - Een object kan wel meerdere keren opgeschreven worden.

Lidmaatschap en notatie van een lidmaatschap

- Object a is **element/lid** van verzameling B : a element van B .
- Object c is **geen element/lid** van A : c niet element van A .
- De **cardinaliteit** van een verzameling is het aantal elementen: $|A|$.
- Een verzameling kan ook een verzameling van verzamelingen zijn.

Set-builder notatie

Set-builder notatie is een andere manier om een verzameling te beschrijven. Bij set-builder notatie wordt een verzameling beschreven.

$V = \{x \mid x \text{ is een priemgetal} \wedge x \leq 10 \wedge x \text{ is een element van } N\}$

$V = 1, 2, 3, 5, 7$

De lege verzameling

Een verzameling zonder elementen/objecten heet een lege verzameling, en wordt aangeduid met een $\emptyset$.

Deelverzamelingen, subsets

Een verzameling is een deelverzameling van een andere als de elementen **voorkomen in beide verzamelingen**.

$A \subset B$ desda voor alle x geldt: als $x \in A$, dan $x \in B$.

$$x \in A \rightarrow x \in B$$

Een strikte deelverzameling is een verzameling waarin de elementen gezamenlijk voorkomen in een van de twee verzamelingen.

$A \subset B$, als geldt: $A \neq B$.

Gelijkheid van verzamelingen

Twee verzamelingen, A en B , zijn **gelijk**, $A = B$, als ze dezelfde elementen hebben. Oftewel, A is een deelverzameling van B EN B is een deelverzameling van A . Als dit niet waar is, zijn de verzamelingen **ongelijk**, oftewel, $A \neq B$.

Inclusie / Deelverzameling-relaties

Het **comprehensieprincipe** zegt dat eigenschappen definiëren verzamelingen. Dit brengt operaties met zich mee voor inclusierelaties.

Naam	Symbool	Wat het betekent
Vereniging	$A \cup B$	Alles wat in A OF in B zit.
Doorsnede	$A \cap B$	Alles wat in A EN in B zit.
Verschil	$A - B$	Alles wat in A zit, en niet in B zit.
Symmetrisch verschil	$A \Delta B, A \text{ xor } B$	Alles wat in A OF in B zit, maar niet in A EN B zit.
Complement	A^c	Alles wat niet in A zit (universum, u).

Machtsverzameling

De machtsverzameling van een verzameling A is de verzameling van alle deelverzamelingen van A.

$$2^A = \{ X \mid X \subseteq A \}$$

Oftewel, als $A = \{A, B\}$, dan is $2^A = \{ \emptyset, \{A\}, \{B\}, \{A, B\} \}$.

Eigenschappen van machtsverzamelingen

- Als $A \subseteq B$, dan $2^A \subseteq 2^B$.
- Als $2^A \subseteq 2^B$, dan $A \subseteq B$.
- Als $2^A = 2^B$, dan $A = B$.

Cartesisch product

Verzamelingen hebben de volgende eigenschap: $\{A, B\} = \{B, A\}$. Geordende paren hebben die eigenschap niet. Dat wil zeggen, voor geordende paren geldt: $(a, b) \neq (b, a)$.

Het cartesisch product $A \times B$ is een verzameling van alle geordende paren (x, y) , zodat x een element is van A en y een element is van B .

$$A \times B =_{def} \{ (x, y) \mid x \in A, y \in B \}$$

$$A = \{a, b, c, d, e, \dots\}$$

$$B = \{1, 2, 3, 4, 5, \dots\}$$

$$A \times B = \{ (a, 1), (a, 2), (a, 3), \dots, (e, 1), (e, 2), \dots \}$$

$$B \times A = \{ (1, a), (2, a), \dots \}$$

Eigenschappen en relaties

Een **eigenschap** van de elementen van A is gedefinieerd als de deelverzameling van A die de objecten van de eigenschap bevat. Een **relatie** tussen de elementen van A en B is gedefinieerd als de deelverzameling van $A \times B$ die de paren objecten bevat waarvoor de relatie geldt.

Functies

Een **functie** is een speciale relatie waar geldt dat voor elke x waarde/input precies een y waarde/output is.

Predicatenlogica

Predicatenlogica is eigenlijk een soort taal waarmee we de **interne structuur** van proposities kunnen analyseren. De predicatenlogica bevat veel soorten **categorieën** van concepten.

Eerste ordetaal

Een eerste ordetaal wordt **ontworpen per toepassing**, oftewel, per probleem/vraag. Dit wordt gedaan door middel van *niet-logische* symbolen.

- Predicatensymbolen: $P_0, P_1, \dots$, > 0 -plaatsig (P, Q, R).
- Functiesymbolen: $F_0, F_1, \dots$, > 0 -plaatsig (f, g).
- Namen: $c_0, c_1, \dots$, 0 -plaatsige functies (a, b, c).
- Variabelen: $x_0, x_1, \dots$, we schrijven x, y, z .
- Connectieven: $\neg, \rightarrow, \leftrightarrow, \wedge, \vee$.
- Kwantoren: $\forall, \exists$.
- Gelijkheidssymbolen: $=$
- Haakjes en de comma.

Voorbeeld van een vertaling van een zin naar een eerste ordetaal

“Als alle studenten slim zijn, halen zij een ≥ 6 voor Redeneren en Logica.”

- | | |
|--|-----------|
| • $S(x)$: x is een student. | PREDICAAT |
| • $I(x)$: x is slim (“intelligent”). | PREDICAAT |
| • t : het vak redeneren en logica. | NAAM |
| • $G(x, y)$: x is groter dan of gelijk aan y . | PREDICAAT |
| • $c(x, y)$: het cijfer van Student(x) voor het vak(y). | FUNCTIE |
| • 6 : het cijfer zes. | NAAM |

$$\forall x (S(x) \rightarrow G(c(x, t), 6))$$

$$(\forall x (S(x) \rightarrow I(x)) \rightarrow \forall x (S(x) \rightarrow G(c(x, t), 6)))$$

Syntax van de predicaatlogica

Een eerste-ordetaal wordt altijd ontworpen **per toepassing/per probleem**. In een eerste-ordetaal onderscheiden we tussen twee **syntactische categorieën**: termen en formules.

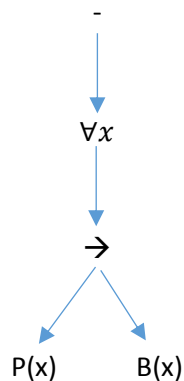
Termen/beroepsjargon van de predicaatlogica

- TERM = Een verzameling met alle *syntactisch gecategoriseerde* **termen** van een eerste ordetaal.
- FORM = Een verzameling met alle *syntactisch gecategoriseerde* **formules** van een eerste ordetaal.
- ATOM = Formules in een van deze vormen:
 - $(t_1 = t_2)$.
 - $P(t_1, \dots, t_n)$.
- VAR = Een verzameling met alle variabelen van een eerste ordetaal.

Ontledingsbomen

Ontledingsbomen zijn bomen waarbij je de formule opbreekt in z'n elementaire delen.

$$\neg \forall x (P(x) \rightarrow B(x))$$



Echte subformule

Een formule A is een subformule van formule F als aan een van de volgende vier voorwaarden wordt voldaan:

- $A = F$
- $F = \neg C$, en A is een subformule van C.
- $F = (C * D)$, en A is een subformule van C.
- $F = Qx C$, en A is een subformule van C.

Een formule A is een **echte subformule** van formule F als A een subformule is van F en $A \neq F$.

Scope

Stel je voor, we hebben: $(\forall x A(x, y) \rightarrow \exists z B(x, z))$.

- De x in $A(x, y)$ is **gebonden** aan de kwantor $\forall x$.
- De y in $A(x, y)$ is **vrij**.
- De x in $B(x, z)$ is **vrij**, niet gebonden aan de kwantor.
- Het **bereik** van $\forall x$ is $A(x, y)$.

Boommethodes voor predicaatlogica

Bij de boommethode voor predicaatlogica gebruiken we dezelfde technieken als voor de boommethode die we al kennen, alleen is er nu een aanvulling op de al bestaande regels.

$\forall$ -regels		$\exists$ -regels		$=$ -regel	sub-regel
$\neg \forall x A$	$\forall x A$	$\neg \exists x A$	$\exists x A$		$s = t$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\exists x \neg A$	$A[x := t]$	$\forall x \neg A$	$A[x := c]$	$t = t$	$A[x := s]$
					$\vdots$
					$A[x := t]$