AUT16

Verificatie van Wiskundige Bewijzen door een Computer

(Een voorstudie ten behoeve van een project AUTOMATH)

Colloquiumvoordracht door Prof.dr. N.G. de Bruijn aan de T.H. Eindhoven, 9 januari 1967.

1. Inleiding.

De AUTOMATH kan een automaat worden die wiskundige stellingen in perfecte vorm met bewijs en al aflevert, mits voortdurend gesouffleerd door een wiskundige. De mate van samenwerking tussen mens en machine die daarbij vereist is laat zich het beste aanvoelen door een vergelijking te maken met de automobiel.

De verwachting bestaat dat de AUTOMATH binnen bestaande computers kan worden gerealiseerd.

In het werken met en door de AUTOMATH zijn vier stadia te onderkennen.

1°. C, de creatieve wiskundige, schept iets. Hij schrijft dit in voor hem heldere taal op (of heeft dat lang geleden reeds gedaan). Wellicht heeft hij een aantal belangrijke details geheel over het hoofd gezien. Het opvullen van deze gaten vraagt vaak veel minder creativiteit dan de hoofdzaak (maar vaak juist veel méér!)

2°. R, de reproductieve competente wiskundige, formuleert stelling en bewijs op voor de moderne wiskundige waterdichte manier. Hij moet het betreffende vakgebied goed kennen, inclusief een aantal wortels ervan. Ook moet hij axiomatisch goed geschoold zijn.

3°. P, de wiskundige programmeur, vertaalt de door zijn voorman geschreven tekst in een stel voor de AUTOMATH verteerbare aanroepen. Hij behoeft de wiskundige inhoud niet te begrijpen, maar moet goed op de hoogte zijn van de mathematische redeneervormen, van de aanroepcodes van AUTOMATH, van de manier waarop die worden verwerkt. Hij moet voldoende ervaring hebben om kleine hiaten in de beschouwingen van R te kunnen constateren en opvullen.

4°. A, de AUTOMATH, beschouwt de aanroepen van P als aanwijzingen voor de constructie van een perfecte tekst. A volgt de aanwijzingen van P slechts op als hij er één van de stappen in herkent die hij volgens zijn eigen normbesef mag nemen. Alles wat A doet is goed. Of het interessant en ter zake is wat A doet, hangt volledig van C, R en P af. Als alles goed verloopt levert A wat R bedoeld heeft. Of het ook is wat C bedoeld heeft, is doorgaans niet meer na te gaan.

De AUTOMATH dwingt ons tot definitieve en genadeloze contrôle van alle wiskundige teksten die we eraan willen onderwerpen. Reeds zonder aan het werk te gaan heeft hij invloed op het doen en laten van de wiskundige. Hij kan een belangrijke rol spelen in de communicatie tussen wiskundigen onderling, in het bijzonder in het publicatiewezen. Een lang bewijs met veel oninteressant detailwerk kan zijn uiteindelijke formulering krijgen in de AUTOMATH-code. De correctheid ervan, die thans vaak een kwestie van geloofwaardigheid is, kan dan ten alle tijde worden vastgesteld.

Het aantal dingen dat de AUTOMATH moet doen is indrukwekkend groot in vergelijking tot de omvang van de oorspronkelijke tekst. Als bij een ijsberg steekt slechts een klein deel ervan boven water in de vorm van de aanroepen. We zullen als voorbeeld een eenvoudige stelling over equivalentieklassen geven (§ 4).

C heeft de Eerste fase geschreven (omstreeks 1930, aan de stijl te zien). R maakte de tweede, en misschien ook de lijst in de kolom UITGESCHREVEN in de derde fase. P schrijft de kolom AANROEP, alsmede de expressies voor zover die door aanroepen zijn ingevoerd (zie de pijltjes). A schrijft de rest van de expressies, houdt de stapel bij, schrijft de kolom BEWIJS, desgewenst de kolom UITGESCHREVEN, en draagt de volledige verantwoordelijkheid.

Doorgaans zal R het het zwaarste hebben, daar gepubliceerd mathematisch werk zeer veel onvolledigheden kan bevatten. Zijn pogingen om tot een voor P zo gemakkelijk mogelijke vorm te komen zouden veelal ook kunnen leiden tot een helderder presentatie van de eerste fase.

Vóórdat een behoorlijke dosis ervaring is verkregen zal het werk voor R zowel als voor P zeer tijdrovend zijn. Men zou het kunnen vergelijken met het leren schrijven van mathematische teksten in het chinees. Na een lange aanleerperiode is het even gemakkelijk als schrijven in de eigen taal. Eén voordeel komt al direct: het schrijven in de vreemde taal dwingt tot duidelijker formulering van de gedachten.

2. Letters en expressies.

Letters. We onderscheiden drie soorten: variabelen, gebonden variabelen, constanten. De gebonden variabelen worden door de automath zelf gekozen; hij doet er geen mededelingen over. De constanten worden door de gebruiker gekozen en ingevoerd met Opal of Opex. De variabelen geven substitutieplaatsen in expressies aan; de gebruiker kan daarvoor de letters kiezen als hij de expressies

zèlf definieert, de automath kan ze naar believen wijzigen. Men bedenke dat de letters x, y, z in een expressiedefinitie als

$$\text{expr. } 225(x,y,z) := (\text{expr. } 127(x,y)) \cap z$$

in het linker- zowel als in het rechterlid variabelen zijn, maar dat ze gebonden zijn in de formule als geheel.

Expressies. Elke variabele is een expressie. Er is verder een lijst met een aantal primitieve expressies, zoals $x \cap y$, $x = y$, $x \in y$. Uit expressies kunnen nieuwe worden gevormd door voor de variabelen expressies te substitueren. Voorbeeld: met $1(x,y)$, $2(u,v)$, $3(p,q)$ kan men vormen

$$4(u,v,p,q) := 1(2(u,v), 3(p,q)) .$$

Ook kan men nieuwe expressies maken uit reeds gevormde door quantizering. Voorbeelden:

$$6(x,y,z) := \forall_{u \in x}\, 5(u,y,z) .$$

$$8(x,y,z) := \{u \in x \mid 7(u,y,z)\} .$$

Er zijn drie typen van expressies: element, set, boolean. Bij elke expressie dient op de een of andere wijze het type te staan aangegeven, zowel als het type van elk van zijn variabelen. Bij substitutie en quantizering zijn er regels t.a.v. deze typen gesteld. Zo mag in $x \in y$ voor x een elementexpressie of setexpressie worden ingevuld, voor y slechts een setexpressie, het resultaat van de substitutie is boolean.

Gevulde expressies ontstaan uit expressies door voor elke variable een constante te substitueren. Alle regels uit een bewijs zijn gevulde boolean expressies. Men kan ook halfgevulde expressies toelaten (waarbij slechts een deel der variabelen door constanten is vervangen); deze kunnen (direct of indirect) slechts dáár gebruikt worden waar de genoemde constanten geldig zijn.

3. Korte beschrijving van een aantal elementaire operaties.

3.1. Algemeen: De regels zijn beschreven met het oog op het samenstellen van formule nr. (k). Een formule (ℓ) heet geldig als $\ell < k$, mits er tussen (ℓ) en (k) geen operaties van een type Afal, Afex, Killex, Afond zijn uitgevoerd. Een letter heet een constante als hij vóór (k) is ingevoerd d.m.v. Opal of Opex en niet inmiddels door Afal, Afex of Killex is afgevoerd.

De automath zorgt er te allen tijde voor dat hij voor gebonden variabelen andere letters gebruikt dan de door de aanroeper ingevoerde, en zal zo nodig de gebonden letters door andere vervangen. De aanroeper hoeft deze letters niet te kennen.

Vele van de hierna genoemde regels hebben een aanroep met slechts één formulenummer. In veel gevallen kan de afspraak worden gemaakt dat dit nummer mag worden weggelaten als het (k - 1) is.

3.2. Stapeloperaties (macro-operaties).

Opal(a,Z). De automath stelt vast dat Z een verzamelingsexpressie is die op legitieme wijze is opgebouwd en waarin alleen geldige letters voorkomen. Hij zet boven op de stapel (a, ∀, (k)) en schrijft als formule

$$a \in Z \ . \qquad (k)$$

Afal a. De automath stelt vast dat er een (a, ∀, (ℓ)) met $\ell < k$ in de stapel ligt, dat boven dit punt van de stapel hoogstens (, ∀, ( ))'s liggen, en dat de laatstgenoemde indertijd niet zijn ingevoerd met behulp van verzamelingsexpressies die a bevatten. Hij schrapt nu (a, ∀, (ℓ)) uit de stapel en schrijft

$$\forall_{y \in X}(Z) \qquad (k)$$

waarbij Z de inhoud van (k - 1) is met a vervangen door y (y is een beschikbare letter), en X het rechterlid van formule (ℓ).

Spaaral a. Als Afal a, maar nu zonder (a, ∀, (ℓ)) uit de stapel te schrappen.

Opex a. De automath ziet dat (k - 1) de vorm heeft van $\exists_{y \in X} B(y, \ldots)$. Hij zet boven de stapel (a, ∃, (k)) en schrijft

$$a \in X \qquad (k)$$

$$B(a, \ldots) \ . \qquad (k+1)$$

In het geval dat (k - 1) slechts luidde $\exists_{y \in X}$, laat hij formule (k + 1) achterwege.

Afex a. De automath stelt vast dat er op de stapel een (a, ∃, (ℓ)) ligt, dat hoger op de stapel geen letters liggen die zijn ingevoerd met verzamelingsexpressies die a bevatten, en dat er hoger op de stapel geen onderstellingen liggen die a bevatten.

Hij schrapt nu (a, ∃, (ℓ)) uit de stapel en schrijft

$$\exists_{y \in X}(Z) \qquad (k)$$

waarbij Z de inhoud van (k - 1) met a vervangen door y is, en X het rechterlid van (ℓ).

Spaarex a. Als Afex a, maar nu zonder (a, ∃, (ℓ)) uit de stapel te schrappen.

Killex a. Als Afex a, maar de automath constateert dat Z de letter a niet bevat, en schrijft

$$Z \qquad (k)$$

Opond Z. De automath constateert dat Z een boolean expressie is die slechts constanten bevat. Hij legt Ond(k) boven op de stapel en schrijft

$$Z \qquad (k)$$

Afond (k). De automath leest Ond (ℓ) boven op de stapel, ziet dat (k - 1) luidt Y, dat (ℓ) luidt Z, schrapt Ond (ℓ) van de stapel en schrijft

$$Z \Rightarrow Y \qquad (k)$$

3.3. Micro-operaties.

3.31. Diverse operaties met wijde strekking.

Subst(j) met (ℓ),(s),(t) (in plaats van dit drietal mogen willekeurig veel formulenummers vermeld worden). De automath constateert dat (ℓ),(s),(t) geldig zijn, en resp. luiden $x_1 \in S_1$, $x_2 \in S_2$, $x_3 \in S_3$ (de x_i's en S_i's zijn expressies), dat (j) geldig is en begint met

$$\forall_{a \in S_1} \forall_{b \in S_2} \forall_{c \in S_3} ;$$

hij copieert formule (j), met weglating van de drie alsymbolen en met vervanging van a, b, c door resp. x_1, x_2, x_3 ; plaatst het resultaat als (k).

String (ℓ),(m),(p). Automath constateert dat (ℓ),(m),(p) geldige formules van het type $a \in b$, $a \subset b$, $a = b$ zijn die achter elkaar geschakeld kunnen worden zoals $a \in b$, $b \subset c$, $c = d$, met als resultaat $a \in d$. Hij levert nu als (k) het resultaat af.

Reverse (ℓ). Als (ℓ) geldt, en luidt resp. $a \subset b$, $a \supset b$, $a = b$, $a \vee b$, $a \wedge b$ dan schrijft de automath als (k) resp. $b \supset a$, $b \subset a$, $b = a$, $b \vee a$, $b \wedge a$.

Opdrachten van het type "exprdef". Hiermee introduceert de gebruiker nieuwe expressies door middel van substitutie of quantizering. De automath accepteert ze pas na contrôle van de expressietypen die erbij betrokken zijn, maar heeft geen bezwaar tegen invoering van een reeds gedefinieerde expressie met een nieuw nummer. Bijv.

exprdef 16(x,y,z) = expr 2(expr 3(x,y), z) ,

exprdef 18(x,y,w) = expr 2(expr 3(x,y), w) ,

exprdef 19(x,y,z) = expr 2(expr 3(x,y), z) .

Opdrachten van het type "exprequal". Voorbeeld:

exprequal expr 7(x,y,z), a, (ℓ), (m)

produceert

expr 7(a,U,P) = expr 7(a,V,Q) (k)

als a een geldige letter is, als (ℓ) de geldige formule U = V is, en (m) de geldige formule P = Q. De automath houdt de typen in het oog.

Opdrachten van het type "tautsubst". Er is een (eventueel door de automaat geautoriseerde) tautologieënlijst. Daar staat bijv. op

Tautologie 24(a,b) := a ∩ b ⊂ a .

Met de opdracht

tautsubst 24(expr 16(x,y), expr 24(x,y))

wordt nu als x en y constanten zijn, geproduceerd (na typecontrôle)

(expr 16(x,y)) ∩ (expr 24(x,y)) ⊂ expr 16(x,y) (k)

3.32. Algemene opmerking over de afleidingsregels Reg 1, ... Bij de aanroep staan steeds één of meer formulenummers vermeld. De automath constateert dat deze formules nog geldig zijn en gaat na of ze de bij de beschouwde afleidingsregels genoemde vorm hebben. In een aantal gevallen komt in zo'n regel een expressie (bijv. Sin regel 4) twee keer voor. De automath controleert dan dat de beide expressies die hij daarvoor verondersteld wordt te substitueren in wezen aan elkaar gelijk zijn. Dit kan betekenen dat in de definities van deze expressies teruggegraven moet worden totdat blijkt dat ze inderdaad gelijk zijn.

De hier genoemde regels staan in de chaotische volgorde waarin ze toevallig in voorbeelden werden ontmoet.

Regel 1 (ℓ).

$$\frac{Z \neq \emptyset \qquad (\ell)}{\exists_{y \in Z} \qquad (k)}$$

waarin y een nieuwe letter is.

Regel 2 (ℓ).

$$\frac{x \in A \cap B \qquad (\ell)}{x \in A \qquad (k)}$$

Regel 3 (ℓ).

$$\frac{x \in A \cap B \qquad (\ell)}{x \in B \qquad (k)}$$

Regel 4 (ℓ), (m).

$$\frac{\begin{array}{ll} x \in S & (\ell) \\ S = \{y \in A \mid B(y)\} & (m) \end{array}}{\begin{array}{ll} x \in A & (k) \\ B(x) & (k+1) \end{array}}$$

Regel 5 (ℓ), (m).

$$\frac{\begin{array}{ll} x \in S & (\ell) \\ S = \{y \in A \mid B(y)\} & (m) \end{array}}{B(x) \qquad (k)}$$

Regel 6 (ℓ), (m), (n).

$$\frac{\begin{array}{ll} S = \{y \in A \mid B(y)\} & (\ell) \\ B(u) & (m) \\ u \in A & (n) \end{array}}{u \in S \qquad (k)}$$

Regel 7 (ℓ),(m).

$$\frac{B_1 \Rightarrow B_2 \quad (\ell) \qquad B_1 \quad (m)}{B_2 \quad (k)}$$

Regel 8 (ℓ),(m).

$$\frac{B_1 \quad (\ell) \qquad B_2 \quad (m)}{B_1 \wedge B_2 \quad (k)}$$

Regel 9 (ℓ),(m).

$$\frac{\forall_{a \in S} \; a \in T \quad (\ell) \qquad \forall_{b \in T} \; b \in S \quad (m)}{S = T \quad (k)}$$

Regel 10 (ℓ),(m).

$$\frac{x \in S \quad (\ell) \qquad x \in T \quad (m)}{x \in S \cap T \quad (k)}$$

Regel 11 (ℓ).

$$\frac{B \Rightarrow \text{contradictie} \quad (\ell)}{\neg B \quad (k)}$$

Regel 12 (ℓ).

$$\frac{A \Rightarrow B \quad (\ell)}{(\neg B) \Rightarrow (\neg A) \quad (k)}$$

Regel 13 (ℓ).

$$\frac{\exists_{y \in A} \; Z \quad (\ell)}{A \neq \emptyset \quad (k)}$$

Regel 14 (ℓ),(m).

$$\frac{A \quad (\ell) \qquad \neg A \quad (m)}{\text{contradictie} \quad (k)}$$

Regel 15 (ℓ).

$$\frac{A \wedge B \quad (\ell)}{A \quad (k) \qquad B \quad (k+1)}$$

Regel 16 (ℓ),(m).

$$\frac{B(a) \quad (\ell) \qquad a \in S \quad (m)}{\exists_{x \in S} \; B(x) \quad (k)}$$

Regel 17 (ℓ).

$$\frac{\exists^{1}_{x \in S} \; B(x) \quad (\ell)}{\exists_{x \in S} \; B(x) \quad (k)}$$

Regel 18 (ℓ),(m),(n),(p),(q).

$$\frac{\begin{array}{ll} \exists^{1}_{x \in S} \; B(x) & (\ell) \\ x \in S & (m) \\ y \in S & (m) \\ B(x) & (p) \\ B(y) & (q) \end{array}}{x = y \quad (k)}$$

Regel 19 (ℓ),(m),(n).

$$\frac{\begin{array}{ll} B(x) & (\ell) \\ x \in S & (m) \\ \forall_{y \in S} \; (B(y) \Rightarrow x = y) & (n) \end{array}}{\exists^{1}_{x \in S} \; B(x) \quad (k)}$$

4. Uitgewerkt voorbeeld

4.1. Eerste fase

Definitie. Een relatie $\equiv$ op een verzameling A heet een equivalentierelatie als zij reflexief, symmetrisch en transitief is, d.w.z. als

(a) $x \equiv x$ voor alle $x \in A$

(b) als $x \equiv y$ dan ook $y \equiv x$

(c) als $x \equiv y$ en $y \equiv z$ dan ook $x \equiv z$.

Definitie. Een deelverzameling σ van A heet een equivalentieklasse als σ bestaat uit alle met eenzelfde element van A equivalente elementen.

Stelling. De equivalentieklassen zijn onderling disjunct; zelfs geldt dat bij twee verschillende equivalentieklassen de elementen van de ene alle

inequivalent zijn met die van de andere.

Bewijs. Laat σ bestaan uit alle met u equivalente elementen, en τ uit alle met v equivalente elementen. Neem aan dat $x \in \sigma$, $y \in \tau$, $x \equiv y$. Dan is $u \equiv x$, $x \equiv y$, $y \equiv v$ (volgens (a) is er geen verschil tussen bijv. $u \equiv x$ en $x \equiv u$). Uit (c) volgt $u \equiv v$. Met dezelfde hulpmiddelen blijkt dat elk met u congruent element ook met v congruent is en omgekeerd. Daaruit volgt $\sigma = \tau$, zodat de tweede uitspraak bewezen is. De eerste volgt daar onmiddellijk uit.

4.2. Tweede fase

Opmerkingen vooraf. De relatie $\equiv$ kan worden beschreven als de verzameling van alle paren $\ulcorner a,b \urcorner$ met $a \in A$, $b \in A$, $a \equiv b$. Deze noemen we S. S is een deel van $A \times A$. De verzameling van alle equivalentieklassen noemen we T. Om hier complicaties t.a.v. de toelaatbaarheid van definities te vermijden doen we alsof T gegeven is en aan (3) voldoet.

Overigens wordt door nauwkeurige analyse van het bewijs een klein winstpuntje geboekt: het blijkt dat de onderstelling der reflexiviteit overbodig is.

Het is niet moeilijk in te zien dat ook $S \subset A \times A$ niet behoeft te worden geëist.

Typedeclaratie: A, S, T zijn verzamelingen.

Onderstellingen:

$$\forall_{x \in A} \forall_{y \in A} (\ulcorner x,y \urcorner \in S) \Rightarrow (\ulcorner y,x \urcorner \in S) \quad , \tag{1}$$

$$\forall_{x \in A} \forall_{y \in A} \forall_{z \in A} (((\ulcorner x,y \urcorner \in S) \wedge (\ulcorner y,z \urcorner \in S)) \Rightarrow \ulcorner x,z \urcorner \in S) \quad , \tag{2}$$

$$T = \{\sigma \in P(A) \mid \exists_{x \in A} (\sigma = \{y \in A \mid \ulcorner x,y \urcorner \in S\})\} \quad . \tag{3}$$

(P(A) is de verzameling van alle deelverzamelingen van A).

Beweringen:

$$\forall_{\sigma \in T} \forall_{\tau \in T} (\sigma \neq \tau \Rightarrow \sigma \cap \tau = \emptyset) \quad ,$$

$$\forall_{\sigma \in T} \forall_{\tau \in T} (\sigma \neq \tau \Rightarrow (\forall_{x \in \sigma} \forall_{y \in \tau} \ulcorner x,y \urcorner \notin S)) \quad .$$

Bewijs. Zij $\sigma \in T$, $\tau \in T$. Er bestaan $a \in A$ en $b \in A$ zó dat σ (resp. τ) de

verzameling is van alle $y \in A$ met $\ulcorner a,y \urcorner \in S$ (resp. $\ulcorner b,y \urcorner \in S$).

Onderstel eens dat $\sigma \cap \tau \neq \emptyset$. Dan is er een $x \in \sigma \cap \tau$, zodat $x \in \sigma$, $x \in \tau$. Daar $x \in \sigma$, is $x \in A$, $\ulcorner a,x \urcorner \in S$. Daar $x \in \tau$, is ook $\ulcorner b,x \urcorner \in S$. Wegens (1) is ook $\ulcorner x,a \urcorner \in S$. Uit $\ulcorner b,x \urcorner \in S$, $\ulcorner x,a \urcorner \in S$ volgt $\ulcorner b,a \urcorner \in S$ met behulp van (2).

Voor iedere $u \in \sigma$ geldt $u \in A$, $\ulcorner a,u \urcorner \in S$. Uit dit laatste en uit $\ulcorner b,a \urcorner \in S$ volgt (weer met (2)) dat $\ulcorner b,u \urcorner \in S$. Wegens $u \in A$, $\ulcorner b,u \urcorner \in S$ is $u \in \tau$.

Voor iedere $u \in \tau$ geldt $u \in A$, $\ulcorner b,u \urcorner \in S$. Reeds is gebleken dat $\ulcorner b,a \urcorner \in S$, met behulp van (1) blijkt nu $\ulcorner a,b \urcorner \in S$. Uit $\ulcorner a,b \urcorner \in S$ en $\ulcorner b,u \urcorner \in S$ vinden we (weer volgens (2)) dat $\ulcorner a,u \urcorner \in S$. Wegens $u \in A$, $\ulcorner a,u \urcorner \in S$ is $u \in \tau$.

Bewezen is nu dat voor elke $u \in \sigma$ geldt $u \in \tau$ en omgekeerd. Derhalve is $\sigma = \tau$. Uit $\sigma \cap \tau \neq \emptyset$ volgt dus $\sigma = \tau$, en daarmee is de eerste bewering bewezen.

We gaan vervolgens uit van de onderstelling $\sigma \neq \tau$. Blijkens het voorafgaande is nu $\sigma \cap \tau = \emptyset$. Neem een $x \in \sigma$, een $y \in \tau$ en onderstel dat $\ulcorner x,y \urcorner \in S$. We zullen hieruit een tegenspraak afleiden.

Wegens $x \in \sigma$ is $x \in A$, $\ulcorner a,x \urcorner \in S$. Wegens $y \in \tau$ is $y \in A$, $\ulcorner b,y \urcorner \in S$. Uit $\ulcorner a,x \urcorner \in S$, $\ulcorner x,y \urcorner \in S$ volgt (wegens (2)) $\ulcorner a,y \urcorner \in S$. Derhalve is $y \in \sigma$. Ook was $y \in \tau$, dus $y \in \sigma \cap \tau$. Dit is in strijd met $\sigma \cap \tau = \emptyset$. Daarmee is de tegenspraak bereikt.

4.3. Derde fase.

Deze is weergegeven op de aangehechte tabellen (5 blz.). De tweede fase is op de voet gevolgd. De stapelstructuur is ook gemakkelijk in de tweede fase aan te geven; dit zou de leesbaarheid van de tweede fase ongetwijfeld ten goede komen.

Terwille van de duidelijkheid zijn A, S, T niet op de stapel vermeld; deze letters zijn gedurende de gehele beschouwing geldig. Men kan ook de onderstellingen (1), (2), (3) met Opond op de stapel zetten, en ze nà (71) en (72) afvoeren. Dan komt de slotconclusie vrij van onderstellingen als een implicatie te voorschijn.

AANROEP	EXPRESSIES	STAPEL	BEWIJS	UITGESCHREVEN	NR
		$(A, S, T, (1), (2), (3)$ weggelaten$)$			
expr 1 (x,y)	$\longrightarrow$ $1(x,y) := \ulcorner x, y \urcorner$			**Stelling.** A, S, T verzamelingen. Dan $((1) \wedge (2) \wedge (3)) \Rightarrow ((7^1) \wedge (7^2))$	
expr 2 (x,y)	$\longrightarrow$ $2(x,y) := 1(x,y) \in S$				
expr 3 (x,y)	$\longrightarrow$ $3(x,y) := 2(x,y) \Rightarrow 2(y,x)$				
expr. 4 (x)	$\longrightarrow$ $4(x) := \forall_{y \in A}\ 3(x,y)$				
expr. 5	$\longrightarrow$ $5 := \forall_{x \in A}\ 4(x)$				
expr. 6 (x,y,z)	$\longrightarrow$ $6(x,y,z) := 2(x,y) \wedge 2(y,z)$				
expr. 7 (x,y,z)	$\longrightarrow$ $7(x,y,z) := 6(x,y,z) \Rightarrow 2(x,z)$				
expr 8 (x,y)	$\longrightarrow$ $8(x,y) := \forall_{z \in A}\ 7(x,y,z)$				
expr 9 (x)	$\longrightarrow$ $9(x) := \forall_{y \in A}\ 8(x,y)$				
expr 10	$\longrightarrow$ $10 := \forall_{x \in A}\ 9(x)$				
geg expr 5			geg 5	$\forall_{x \in A} \forall_{y \in A} (\ulcorner x, y \urcorner \in S \Rightarrow \ulcorner y, x \urcorner \in S)$	(1)
geg expr 10			geg 10	$\forall_{x \in A} \forall_{y \in A} \forall_{z \in A} ((\ulcorner x, y \urcorner \in S \wedge \ulcorner y, z \urcorner \in S) \Rightarrow \ulcorner x, z \urcorner \in S)$	(2)
expr. 11 (x)	$\longrightarrow$ $11(x) := \{ y \in A \mid 2(x,y) \}$				
expr 12 (x, σ)	$\longrightarrow$ $12(x,\sigma) := \sigma = 11(x)$				
expr. 13 (σ)	$\longrightarrow$ $13(\sigma) := \exists_{x \in A}\ 12(x,\sigma)$				
expr 14	$\longrightarrow$ $14 := P(A)$				
expr 15	$\longrightarrow$ $15 := \{ \sigma \in 14 \mid 13(\sigma) \}$				
geg. expr 16	$\longrightarrow$ $16 := T = 15$		geg 16	$T = \{ \sigma \in P(A) \mid \exists_{x \in A} (\sigma = \{ y \in A \mid \ulcorner x, y \urcorner \in S \}) \}$	(3)

AANROEP	EXPRESSIES	STAPEL	BEWIJS	UITGESCHREVEN	
Opal (σ, T)		$\sigma, \forall, (4)$	$\varepsilon(\sigma, T)$	$\sigma \in T$	(4)
String (4), (3)	$101(\sigma) := \varepsilon(\sigma, 15)$		$101(\sigma)$	$\sigma \in \{\lambda \in P(A) \mid \exists_{x \in A}(\lambda = \{y \in A \mid \ulcorner x, y \urcorner \in S\})\}$	(5)
Opal (τ, T)		$\tau, \forall, (6)$	$\varepsilon(\tau, T)$	$\tau \in T$	(6)
String (6), (3)	$201(\sigma) := \varepsilon(\sigma, 15)$		$201(\sigma)$	$\tau \in \{\lambda \in P(A) \mid \exists_{x \in A}(\lambda = \{y \in A \mid \ulcorner x, y \urcorner \in S\})\}$	(7)
Reg 5 (5), (3)			$13(\sigma)$	$\exists_{a \in A}(\sigma = \{y \in A \mid \ulcorner a, y \urcorner \in S\})$	(8)
Opex a		$a, \exists, (9)$	$\varepsilon(a, A)$	$a \in A$	(9)
			$12(a, \sigma)$	$\sigma = \{y \in A \mid \ulcorner a, y \urcorner \in S\}$	(10)
Reg 5 (7), (3)			$13(\tau)$	$\exists_{b \in A}(\tau = \{y \in A \mid \ulcorner b, y \urcorner \in S\})$	(11)
Opex b		$b, \exists, (12)$	$\varepsilon(b, A)$	$b \in A$	(12)
			$12(b, \tau)$	$\tau = \{y \in A \mid \ulcorner b, y \urcorner \in S\}$	(13)
expr $17(\lambda, \mu) \longrightarrow$	$17(\lambda, \mu) := \lambda \cap \mu$				
expr $18(\lambda, \mu) \longrightarrow$	$18(\lambda, \mu) := 17(\lambda, \mu) \neq \emptyset$				
Opond $18(\sigma, \tau)$		Ond (14)	$18(\sigma, \tau)$	$\sigma \cap \tau \neq \emptyset$	(14)
Reg 1	$102(\lambda, \mu) := \exists_{u \in 17(\lambda, \mu)}$		$102(\sigma, \tau)$	$\exists_{u \in \sigma \cap \tau}$	(15)
Opex x	$103(\lambda, \mu, x) := \varepsilon(x, 17(\lambda, \mu))$	$x, \exists, (16)$	$103(\sigma, \tau, x)$	$x \in \sigma \cap \tau$	(16)
Reg 2			$\varepsilon(x, \sigma)$	$x \in \sigma$	(17)
Reg 4 (17), (10)			$\varepsilon(x, A)$	$x \in A$	(18)
			$2(a, x)$	$\ulcorner a, x \urcorner \in S$	(19)
Reg 3 (16)			$\varepsilon(x, \tau)$	$x \in \tau$	(20)
Reg 5 (20), (13)			$2(b, x)$	$\ulcorner b, x \urcorner \in S$	(21)

AANROEP	EXPRESSIES	BEWIJS	UITGESCHREVEN	
Subst (1) met (9), (18)		3(a,x)	$\ulcorner a,x \urcorner \in S \Rightarrow \ulcorner x,a \urcorner \in S$	(22)
Reg 7 (22), (19)		2(x,a)	$\ulcorner x,a \urcorner \in S$	(23)
Reg 8 (21) (23)	$106(b,x,a) := 2(b,x) \wedge 2(x,a)$	106(b,x,a)	$\ulcorner b,x \urcorner \in S \wedge \ulcorner x,a \urcorner \in S$	(24)
Subst (2) met (12)(18)(9)		7(b,x,a)	$\left(\left(\ulcorner b,x \urcorner \in S\right) \wedge \left(\ulcorner x,a \urcorner \in S\right)\right) \Rightarrow \ulcorner b,a \urcorner \in S$	(25)
Reg 7 (25), (24)		2(b,a)	$\ulcorner b,a \urcorner \in S$	(26)
Opal (u, σ)		ε(u,σ)	$u \in \sigma$	(27)
Reg 4 (27), (10)		ε(u,A)	$u \in A$	(28)
		2(a,u)	$\ulcorner a,u \urcorner \in S$	(29)
Reg 8 (26), (29)	$116(b,a,u) := 2(b,a) \wedge 2(a,u)$	116(b,a,u)	$\ulcorner b,a \urcorner \in S \wedge \ulcorner a,u \urcorner \in S$	(30)
Subst (2) met (12)(9)(28)		7(b,a,u)	$\left(\left(\ulcorner b,a \urcorner \in S\right) \wedge \left(\ulcorner a,u \urcorner \in S\right)\right) \Rightarrow \ulcorner b,u \urcorner \in S$	(31)
Reg 7 (31) (30)		2(b,u)	$\ulcorner b,u \urcorner \in S$	(32)
Reg 6 (13), (32), (28)		ε(u,τ)	$u \in \tau$	(33)
Afal u	$104(\lambda,\mu) := \forall_{u \in \lambda}\, \varepsilon(u,\mu)$	104(σ,τ)	$\forall_{u \in \sigma}\, u \in \tau$	(34)
Opal (u, τ)		ε(u,τ)	$u \in \tau$	(35)
Reg 4 (35), (12)		ε(u,A)	$u \in A$	(36)
		2(b,u)	$\ulcorner b,u \urcorner \in S$	(37)
Subst (1) met (12), (9)		3(b,a)	$\ulcorner b,a \urcorner \in S \Rightarrow \ulcorner a,b \urcorner \in S$	(38)
Reg 7 (38), (36)		2(a,b)	$\ulcorner a,b \urcorner \in S$	(39)
Reg 8 (39), (37)	$126(x,y,z) := 2(x,y) \wedge 2(y,z)$	126(a,b,u)	$\ulcorner a,b \urcorner \in S \wedge \ulcorner b,u \urcorner \in S$	(40)

STAPEL

σ τ a b (14) x

u, ∀, (27)

u, ∀, (35)

AANROEP	EXPRESSIES	STAPEL σ τ a b (14) x u	BEWYS	UITGESCHREVEN	NR
Subst (2) met (9) [12] (36)			$7(a,b,u)$	$((\ulcorner a,b\urcorner \in S) \wedge (\ulcorner b,u\urcorner \in S)) \Rightarrow (\ulcorner a,u\urcorner \in S)$	(41)
Regel 7 (41), (40)			$2(a,u)$	$\ulcorner a,u\urcorner \in S$	(42)
Reg 6 (10). (42), (36)			$\varepsilon(u,\sigma)$	$u \in \sigma$	(43)
Afal u	$114(\lambda,\mu) := \forall_{u\in\lambda}\ \varepsilon(u,\mu)$	→	$114(\tau,\sigma)$	$\forall_{u\in\tau}\ u\in\sigma$	(44)
Reg 9 (34), (33)			$\gamma(\sigma,\tau)$	$\sigma=\tau$	(45)
Killer x		→	$\gamma(\sigma,\tau)$	$\sigma=\tau$	(46)
Afond (14)	$105(\lambda,\mu) := 18(\lambda,\mu) \Rightarrow \gamma(\lambda,\mu)$	→	$105(\sigma,\tau)$	$(\sigma\cap\tau\neq\emptyset) \Rightarrow (\sigma=\tau)$	(47)
Reg 12	$107(\lambda,\mu) := \neg\ 18(\lambda,\mu)$				
	$108(\lambda,\mu) := \neg\ \gamma(\lambda,\mu)$				
	$109(\lambda,\mu) := 108(\lambda,\mu) \Rightarrow 107(\lambda,\mu)$		$109(\sigma,\tau)$	$(\sigma\neq\tau) \Rightarrow (\sigma\cap\tau=\emptyset)$	(48)
Opond $19(\sigma,\tau)$	$19(\sigma,\tau) := \neg\ \gamma(\sigma,\tau)$	Ond (49)	$19(\sigma,\tau)$	$\sigma\neq\tau$	(49)
Reg 7 (48), (49)			$107(\sigma,\tau)$	$\sigma\cap\tau=\emptyset$	(50)
Opal (x,σ)		x, ∀, (51)	$\varepsilon(x,\sigma)$	$x\in\sigma$	(51)
Reg 4 (51), (10)			$\varepsilon(x,A)$	$x\in A$	(52)
			$2(a,x)$	$\ulcorner a,x\urcorner \in S$	(53)
Opal (y,τ)		y, ∀, (54)	$\varepsilon(y,\tau)$	$y\in\tau$	(54)
Reg 4 (54), (13)			$\varepsilon(y,A)$	$y\in A$	(55)
			$2(b,y)$	$\ulcorner b,y\urcorner \in S$	(56)
Opond $2(x,y)$		Ond (57)	$2(x,y)$	$\ulcorner x,y\urcorner \in S$	(57)

AANROEP	EXPRESSIES	BEWIJS	UITGESCHREVEN	
		STAPEL: $\sigma\ \tau\ a\ b(49)\ x\ y(57)$		
Reg 8 (53),(57)	$136(x,y,z) := 2(x,y) \wedge 2(y,z)$	$136(a,x,y)$	$(\ulcorner a,x\urcorner \in S) \wedge (\ulcorner x,y\urcorner \in S)$	(58)
Subst(2) met (9),(52),(55)		$7(a,x,y)$	$((\ulcorner a,x\urcorner \in S) \wedge (\ulcorner x,y\urcorner \in S)) \Rightarrow \ulcorner a,y\urcorner \in S$	(59)
Reg 7 (59),(58)		$2(a,y)$	$\ulcorner a,y\urcorner \in S$	(60)
Reg 6 (10),(60),(55)		$\varepsilon(y,\sigma)$	$y \in \sigma$	(61)
Reg 10 (61),(54)	$117(\lambda,\mu) := \lambda \cap \mu$			
	$118(\lambda,\mu,x) := x \in 117(\lambda,\mu)$	$118(\sigma,\tau,y)$	$y \in \sigma \cap \tau$	(62)
Spaerex y	$119(\lambda,\mu) := \exists_{x\in\lambda}\ 118(\lambda,\mu,x)$	$119(\sigma,\tau)$	$\exists_{u\in\tau}\ u \in \sigma \cap \tau$	(63)
Reg 13	$120(\lambda,\mu) := 117(\lambda,\mu) \neq \emptyset$	$120(\sigma,\tau)$	$\sigma \cap \tau \neq \emptyset$	(64)
Reg 14 (50),(64)		contr.	contradictie	(65)
Afond (57)	$121(x,y) := 2(x,y) \Rightarrow$ contr	$121(x,y)$	$\ulcorner x,y\urcorner \in S \Rightarrow$ contradictie	(66)
Reg 11	$122(x,y) := \neg\ 2(x,y)$	$122(x,y)$	$\ulcorner x,y\urcorner \notin S$	(67)
Afal y, Afal x, Afon (5)	$123(x,\mu) := \forall_{y\in\mu}\ 122(x,y)$			
	$124(\lambda,\mu) := \forall_{x\in\lambda}\ 123(x,\mu)$			
	$125(\lambda,\mu) := 119(\lambda,\mu) \Rightarrow 124(\lambda,\mu)$	$125(\sigma,\tau)$	$(\sigma \neq \tau) \Rightarrow \forall_{x\in\sigma} \forall_{y\in\tau}\ \ulcorner x,y\urcorner \notin S$	(68)
Killex b, Killex a (68)(48)		$125(\sigma,\tau)$	$(\sigma \neq \tau) \Rightarrow \forall_{x\in\sigma} \forall_{y\in\tau}\ \ulcorner x,y\urcorner \notin S$	(69)
		$109(\sigma,\tau)$	$(\sigma \neq \tau) \Rightarrow (\sigma \cap \tau = \emptyset)$	(70)
Afex τ Afex σ (70) (69)	$127(\lambda) := \forall_{\mu\in T}\ 109(\lambda,\mu)$ $128 := \forall_{\lambda\in T}\ 127(\lambda)$.	128	$\forall_{\sigma\in T} \forall_{\tau\in T} ((\sigma \neq \tau) \Rightarrow (\sigma \cap \tau = \emptyset))$	(71)
	$129(\lambda) := \forall_{\mu\in T}\ 125(\lambda,\mu)$ $130 := \forall_{\lambda\in T}\ 129(\lambda)$	130	$\forall_{\sigma\in T} \forall_{\tau\in T} ((\sigma \neq \tau) \Rightarrow (\forall_{x\in\sigma} \forall_{y\in\tau}\ \ulcorner x,y\urcorner \notin S))$	(72)