De stelling van Church

Bachelorscriptie
Departement Wiskunde
Universiteit Utrecht

Auteur: Maartje Bongers
Begeleider: Dr. Jaap van Oosten

April 2016 - Juni 2016

Contents

1 Inleiding

Logica heeft een grote invloed gehad, en heeft dit nog steeds, op wiskunde. De belangrijkste bijdrage die logica heeft geleverd binnen de wiskunde, is het geven van een formele definitie voor bewijzen en voor algoritmische berekenbaarheid. De echte opkomst van logica in de wiskunde begon pas in het begin van de 20e eeuw. Rond deze tijd hielden bijna alle bekende wiskundige zich bezig met logica in de wiskunde. Van de 23 nog niet opgeloste problemen die de Duitse wiskundige David Hilbert in 1900 naar het Internationale Congres voor Wiskundige stuurde, gingen de eerste twee problemen letterlijk over logica. Dit geeft wel aan hoe belangrijk logica in die tijd in de wiskunde was. David Hilbert is hiernaast ook de bedenker van het "Entscheidungsproblem". Het "Entscheidungsproblem" is het volgende vraagstuk: Bestaat er een algoritme om te bepalen of een zin ϕ bewijsbaar is, waarbij de zin ϕ een eerste-orde logische formule is. Zowel Alonzo Church als Alan Turing kwamen in 1936 met hun artikelen, respectievelijk [1] en [2], die dit vraagstuk negatief beantwoorden. Dus dat er geen algoritme bestaat dat iedere logische zin bewijst. Het bewijs dat Church hiervoor geeft, is de stelling van Church en hier zullen we het deze scriptie verder over hebben.

In deze scriptie zullen we eerst de primitief recursieve functies behandelen. Daarna zullen we ons verdiepen in de Peano Rekenkunde om uiteindelijk met deze twee onderwerpen gecombineerd een bewijs te kunnen leveren.

Er wordt verwacht dat de lezer bekend is met eerste-orde logica. Mocht dit niet het geval zijn, dan raad ik aan om hoofdstuk 1 en 2 van [3] te bestuderen, voordat men aan deze scriptie begint.

1.1 Bronvermelding

Deze scriptie is gebaseerd op de dictaten [3] en [4] van Dr. Jaap van Oosten, waarbij dictaat [3] duidelijk de boventoon voert. Wanneer er gebruik is gemaakt van andere bronnen staat dit vermeld.

2 Recursieve functies

2.1 Inleiding

De primitief recursieve functies vormen een deelklasse van de berekenbare functies. Een kenmerk van primitief recursieve functies is dat ze totaal en berekenbaar zijn, maar andersom geldt dit niet. Dit wil dus zeggen dat niet alle totale, berekenbare functies primitief recursief zijn. Een voorbeeld hiervan is de Ackermannfunctie. De klasse van functies waarvoor wel geldt dat primitief recursieve functies totaal en berekenbaar zijn en dat als een functie totaal en berekenbaar is deze ook recursief is, noemen we de μ-recursieve fucties, ook wel partieel recursieve functies. Hier komen we in de volgende paragraaf op terug.

2.2 Primitief recursieve functies

Primitief recursieve functies zijn functies van natuurlijke getallen. Dit wil zeggen dat we een of meerdere getallen $\in \mathbb{N}$ in de functie invoeren en als uitkomst ook een natuurlijk getal krijgen. De primitief recursieve functies vormen de kleinste verzameling functies die voldoet aan de voorwaarden hieronder. Voor de definitie van primitief recursieve functies en de voorbeelden die in deze paragraaf volgen, is er naast [3] ook gebruik gemaakt van [5].

- De nulfunctie. De functie Z: $\mathbb{N} \to \mathbb{N}$ waarvoor geldt $Z(x) = 0$ voor iedere $x \in \mathbb{N}$, is primitief recursief.

- De opvolgersfunctie. De functie S: $\mathbb{N} \to \mathbb{N}$ waarvoor geldt $S(x) = x+1$ voor iedere $x \in \mathbb{N}$, is primitief recursief.

- De projectiefunctie. Voor iedere $j \geq 1$ en i met $1 \leq i \leq j$ is de projectiefunctie $\prod_i^j$: $\mathbb{N}^j \to \mathbb{N}$ gedefineerd als $\prod_i^j(x_1, \cdots, x_j) = x_i$, is primitief recursief. Voor iedere $x_1, \cdots, x_i \in \mathbb{N}$.

- De compositie. De compositie van primitief recursieve functies is primitief recursief. De compositie is als volgt gedefinieerd. Gegeven f een primitief recursieve functie in n argumenten en $g_1, \cdots, g_n$ primitief recursieve functies in m argumenten dan geldt dat

  $$f(g_1(x_1, \cdots, x_m), \cdots, g_n(x_1, \cdots, x_m))$$

  een primitief recursieve functie is in m argumenten voor iedere $x \in \mathbb{N}$.

- Primitieve recursie. We hebben twee functies F en G die primitief recursief zijn in respectievelijk n en $n+2$ argumenten. Dat wil zeggen $F : \mathbb{N}^n \to \mathbb{N}$ en $G : \mathbb{N}^{n+2} \to \mathbb{N}$. Dan geldt voor de functie H dat deze ook primitief recursief is, maar dan in $n+1$ argumenten. Functie H is als volgt gedefinieerd:

$$H(0, (x_1, \cdots, x_n)) = F(x_1, \cdots, x_n)$$
$$H(y+1, (x_1, \cdots, x_n)) = G(y, H(y, (x_1, \cdots, x_n)), (x_1, \cdots, x_n))$$

Bij het laatste puntje van de gegeven voorwaarden, sluiten we het geval waarbij $n = 0$ niet uit. In dat geval zeggen we dat $G : \mathbb{N}^2 \to \mathbb{N}$ is primitief recursief en $k \in \mathbb{N}$, dat dan de functie H, gedefineerd door:

$$H(0) = k$$
$$H(y+1) = G(y, H(y))$$

primitief recursief is.

Wanneer we van een k-plaatsige relatie spreken, dan bedoelen we een deelverzameling van $\mathbb{N}^k$. We zullen gebruik maken van de volgende afspraak voor de karakteristieke functie $\chi_A : \mathbb{N}^k \to \mathbb{N}$ voor de k-plaatsige relatie A:

$$\chi_A(\vec{x}) \;=\; \begin{cases} 0 & \text{als } x \in A \\ 1 & \text{anders} \end{cases}$$

We zeggen dat een functie primitief recursief is, als zijn karakteristieke functie primitief recursief is.

Omdat dit allemaal nog niet tot de verbeelding spreekt, volgen nu een aantal voorbeelden van recursieve functies.

Voorbeeld 1:
Stel we hebben de functie $f(x) = x + 5$. Dan kunnen we formeel zeggen dat

1. $S(x) = x + 1$ een primitief recursieve functie is volgens de opvolgersfunctie.

2. $S(S(x))$ een primitief recursieve functie is volgens de compositieregel. Hierbij gebruiken we de functie gevonden bij 1 samengesteld met de functie gedefinieerd in 1.

3. $S(S(S(x)))$ een primitief recursieve functie is volgens de compositieregel. Hierbij gebruiken we de functie gevonden in 2 samengesteld met de functie gedefinieerd in 1.

4. $S(S(S(S(x))))$ een primitief recursieve funcite is volgens de compositieregel. Hierbij gebruiken we de functie gevonden in 3 samengesteld met de functie gedefinieerd in 1.

5. $S(S(S(S(S(x)))))$ een primitief recursieve functie is volgends de compositieregel. Hierbij gebruiken we de functie gevonden in 4 samengesteld met de functie gedefinieerd in 1.

Informeel kunnen we nu zeggen dat $f(x) = S(S(S(S(S(x)))))$ een primitief recursieve functie is volgends de opvolgersfunctie en herhaaldelijk toepassen van de compositieregel. Merk op dat we de compositieregel, zoals we die hierboven hebben uitgevoerd, willekeurig vaak kunnen toepassen en steeds een primitief recursieve functie overhouden.

Voorbeeld 2:
Stel we hebben de functie $f(x, y) = x + y$. Dan kunnen we formeel zeggen dat:

1. $g(x) = \prod_1^1(x) = x$ een primitief recursieve functie is volgens de projectiefunctie.

2. $\prod_3^3(x, y, z) = z$ een primitief recursieve functie is volgens de projectiefunctie.

3. $S(x) = x + 1$ een primitief recursieve functie is volgens de opvolgersfunctie.

4. $h(x, y, z) = S(\prod_3^3(x, y, z)) = z + 1$ volgens de compositie. Hierbij zijn S en $\prod_3^3$ samengesteld.

5. We definiëren nu f volgens de recursie en we gebruiken g, h. We hebben dat

$$f(x, 0) = g(x) = \prod_1^1(x) = x$$
$$f(x, n + 1) = h(x, n, f(x, n)) = S(\prod_3^3(x, n, f(x, n)) = f(x, n) + 1$$

   Informeel kunnen we dus zeggen dat

$$f(x, 0) = x$$
$$f(x, n + 1) = f(x, n) + 1.$$

We hebben hier laten zien dat plus primitief recursief is. Dit gaan we nu ook doen voor vermenigvuldigen, met behulp van wat we net hebben laten zien.

Voorbeeld 3:
De functie $f = xy$ is primitief recursief via

$$f(x,0) = 0$$
$$f(x,y+1) = \prod_3^3(y, f(x,y), x) + \prod_2^3(y, f(x,y), x) = x + f(x,y)$$

We hebben hiermee aangetoond dat vermenigvuldiging primitief recursief is. Dit gaan we gebruiken om ook te laten zien dat machtsverheffing primitief recursief is.

Voorbeeld 4:
De functie $f = x^y$ is primitief recursief via

$$f(x,0) = 1$$
$$f(x,y+1) = \prod_3^3(y, f(x,y), x)\prod_2^3(y, f(x,y), x) = xf(x,y)$$

Hiermee hebben we aangetoond dat ook functies met machtsverheffingen primmitief recursief zijn.
Het lijkt nu logisch om te laten zien dat functies met aftrekken ook primitief recursief zijn. Hier hebben we echter een probleem, primitief recursieve functies zijn namelijk gedefinieerd van $\mathbb{N} \to \mathbb{N}$ en dat houdt dus in dat ze niet negatief kunnen zijn. We introduceren nu het begrip 'afgeknopte aftrekking' ook wel 'monus' genoemd.

Voorbeeld 5:
In alle andere voorbeelden die we tot nu toe hebben gezien, hadden we te maken met een bekende functie waarvan we moesten bewijzen dat deze primitief recursief is. In dit voorbeeld introduceren we direct een functie. De functie werkt als volgt: als $x \geq 0$ dan wordt er 1 van de functie afgetrokken, anders blijft de functie 0. Dus:

$$f(0) = 0$$
$$f(x+1) = x$$

Merk op dat in de recursie de functie h gedefineerd kan worden door zowel x en $f(x)$. We noemen deze functie $P(x)$. We gaan P gebruiken om een versie van aftrekken te definiëren. P komt van predecessor, in het Nederlands voorganger.

Voorbeeld 6:
Stel we hebben de functie $f(x,y) = x - y$. Dat wil zeggen, $x - y$ als $x - y \geq 0$ anders 0.

$$f(x,0) = 0$$
$$f(x,y+1) = P(f(x,y))$$

2.3 Partieel recursieve functies

Voordat we beginnen met de partieel recursieve functies, moeten we eerst het begrip "partiële functie" definiëren.

Definitie 7
Laat X en Y twee verzamelingen zijn. Een partiële functie F van X naar Y is een functie $F : U \to Y$ waarbij U een deelverzameling van X is. We noemen U dan het domein van F en dat noteren we als volgt: $\mathrm{dom}(F)$. We schrijven $F : X \rightharpoonup Y$ om aan te geven dat F een partiële functie is van X naar Y. De functie F is totaal als $\mathrm{dom}(F) = X$. Totale functies zijn een een speciale groep binnen de partiële functies. Een partiële functie is soms totaal, maar dus niet altijd.

Als $x \in X$, zeggen we dat $F(x)$ gedefinieerd is als $x \in \mathrm{Dom}(F)$. Partiële functies kunnen ook worden samengevoegd. Als we namelijk de functies $F : X \rightharpoonup Y$ en $G : Y \rightharpoonup Z$ hebben, dan is $GF : X \rightharpoonup Z$ de functie met als domein de deelverzameling $\{x \in X | x \in \mathrm{dom}(F) \text{ en } F(x) \in \mathrm{dom}(G)\}$ van X.

We gebruiken ook het symbool $\simeq$ (Kleene equality) tussen uitdrukkingen $F(x)$ en $G(x)$ voor partiële functies. $F(x) \simeq G(x)$ betekent dat $F(x)$ gedefinieerd is precies wanneer $G(x)$ gedefinieerd is en als dit het geval is, geldt: $F(x) = G(x)$. Dus $F(x) \simeq G(x)$ geldt ook wanneer ze allebei niet gedefinieerd zijn. Merk op dat een term enkel gedefinieerd kan zijn, als alle subtermen ook gedefinieerd zijn. Een voorbeeld hiervan: Als $\prod_1^2$ wijst op de eerste projectie van $\mathbb{N}^2 \to \mathbb{N}$, zoals we eerder gedefinieerd hebben en $G : \mathbb{N} \rightharpoonup \mathbb{N}$ is een partiële functie dan geldt dat $\prod_1^2(x, G(y))$ alleen gedefinieerd is als $G(y)$ gedefinieerd is, en dus $\prod_1^2(x, G(y)) \simeq x$ hoeft niet te gelden.

De klasse van partieel recursieve functies $\mathbb{N}^k \rightharpoonup N$, voor een variabele k, voldoet aan de volgende voorwaarden:

- Alle primitief recursieve functies zijn partieel recursief.
- De partieel recursieve functies zijn gesloten onder de volgende compositie. Als $G_1, \cdots, G_l : \mathbb{N}^k \rightharpoonup \mathbb{N}$ en $H : \mathbb{N}^l \rightharpoonup \mathbb{N}$ partieel recursief zijn, dan geldt ook dat de functie $H(G_1(x_1, \cdots, x_k), \cdots, G_l(x_1, \cdots, x_k))$ partieel recursief is. Deze functie is gedefinieerd voor alle $(x_1, \cdots, x_k) \in \cap_{i=1}^{l} \mathrm{dom}(G_i)$ waarvoor geldt:

$$(G_1(x_1, \cdots, x_k), \cdots, G_l(x_1, \cdots, x_k)) \in Dom(H)$$

- Als $G : \mathbb{N}^{k+1} \rightharpoonup \mathbb{N}$ partieel recursief is, dan is ook F partieel recursief. Waarbij F gedefinieerd is uit G door minimalisatie. We zeggen:

  $$F(x_1, \cdots, x_k) \simeq G((x_1, \cdots, x_k), y) = 0$$

  voor iedere $y \in \mathbb{N}$.

  $F(x_1, \cdots, x_k)$ is gedefinieerd dan en slechts dan als er een y bestaat zodat $\forall i \leq y$ het deel $((x_1, \cdots, x_k), i)$ in het domein van G zit en dat $G(x_1, \cdots, x_k) = 0$. $F(x_1, \cdots, x_k)$ geeft dan de kleinste y met de boven genoemde eigenschap.

Een relatie $A \subset \mathbb{N}^k$ is recursief als zijn karakteristieke functie χ_A partieel recursief is. We hebben χ_A al gedefinieerd bij de primitief recursieve functies. Een partieel recursieve functie is totaal recursief, ook wel recursief, als deze totaal is. Omdat χ_A altijd een totale functie is voor iedere relatie A, bestaat er geen begrip zoals 'partieel recursieve relatie'.

2.4 *Smn*-theorem

Rond 1930 verdiepten logici zich in de vraag wat een berekenbare functie precies is: een functie (mogelijk partieel) $F : \mathbb{N}^k \rightharpoonup \mathbb{N}$ waarvoor een algoritme bestaat dat het voor mensen of computers mogelijk maakt om stap voor stap de waarde van F te berekenen, voor gegeven argumenten. Er zijn mogelijk ook argumenten waarvoor het algoritme nooit een eindstatus bereikt.
We moeten dus eerst weten wat een algoritme is voordat we verder kunnen met berekenbare functies. Verschillende logici hebben zich hiermee bezig gehouden en het bleek dat ze allemaal dezelfde definitie hadden gevonden voor een partieel berekenbare functie; namelijk de partieel recursieve functie. We geven nu een aantal stellingen, die gebruikt worden bij partieel recursieve functies, maar voordat we dat doen moeten we nog een definitie introduceren die betrekking heeft op bijecties: $\mathbb{N}^n$ met $n > 2$. In het algemeen worden bijecties gegeven met een polynomiale macht van n, maar wij zullen gebruik maken van een hogere macht.

Definitie 8
De bijecties $j^m : \mathbb{N}^m \to \mathbb{N}$ voor $m \geq 1$ zijn als volgt gedefinieerd:

$$j^1 \text{ is de identiteitsfunctie}$$
$$j^{m+1}(x_1, \ldots, x_m, x_{m+1}) = j(j^m(x_1, \ldots x_m), x_{m+1}).$$

Dan hebben we daarnaast ook nog de projectiefuncties $j_i^m : \mathbb{N} \to \mathbb{N}$ voor $1 \leq i \leq m$ die voldoet aan:

$$j^m(j_1^m(z), \ldots, j_m^m(z)) = z$$

voor iedere $z \in \mathbb{N}$. En daarnaast geldt nog dat

$$j_1^1(z) = z$$

$$j_1^{m+1}(z) \;=\; \begin{cases} j_i^m(j_1(z)) & \text{als } 1 \leq i \leq m \\ j_2(z) & \text{als } i = m+1 \end{cases}$$

Stelling 9 (Kleene Enumeration Theorem)
Er is een 4-plaatsige primitief recursieve relatie T en een unaire primitief recursieve functie U zodat voor iedere partieel recursieve functie $F : \mathbb{N}^k \rightharpoonup \mathbb{N}$ er een nummer e bestaat, met de volgende eigenschappen (e is de index van de functie F):

- Voor alle k-tallen $n_1, \cdots, n_k$ hebben we dat $F(n_1, \cdots, n_k)$ gedefinieerd is dan en slechts dan als er een nummer y is zodat $T(k, e, j^k(n_1, \cdots, n_k), y)$ bestaat. Dit houdt in dat $(k, e, j^k(n_1, \cdots, n_k), y) \in T$.
- Als $F(n_1, \cdots, n_k)$ gedefinieerd is, dan $F(n_1, \cdots, n_k) = U(y)$ voor de kleinste y, zoals gedefinieerd bij het puntje hierboven.

Als e overeenkomt met de k-plaatsige partieel recursieve functie F, zoals gedefinieerd in Stelling 9, dan hebben we:

$$F(n_1, \cdots, n_k) \simeq U(T(k, e, j^k(n_1, \cdots, n_k), y))$$

en dit schrijven we als $\varphi_e^{(k)}$ voor F.
De letters T en U worden standaard gebruikt in de berekenbaarheidstheorie. De relatie T wordt ook wel het **Kleene T-predikaat** genoemd en U wordt dan de **uitkomstfunctie** genoemd. Omdat de relatie T primitief recursief is, is de partiële functie, $\Psi(m, e, x) \simeq U(T(m, e, x, y))$ partieel recursief. Dit houdt dus ook in dat iedere k-plaatsige partieel recursieve functie van de vorm $\Psi(k, e, j^k(x_1, \cdots, x_k))$ is, voor een zekere e. Een algoritme van de vorm Ψ wordt daarom ook wel een universeel algoritme genoemd. Er bestaat in tegenstelling tot wat we net hebben laten zien, geen universeel algoritme voor totaal recursieve functies, zie het bewijs hieronder.

Stelling 10
Er bestaat geen totaal recursieve functie $\Psi(m, e, x)$ zodat iedere totaal recursieve functie $F : \mathbb{N}^k \to \mathbb{N}$ gelijk is aan $\Psi(k, e, j^k(x_1, \cdots, x_k))$ voor een zekere e.

Bewijs
We nemen het tegenovergestelde aan, dus neem aan dat zo'n functie Ψ bestaat. Dan zou moeten gelden dat de functie

$$\Psi(k, j^k(x_1, \cdots, x_k), j^k(x_1, \cdots, x_k)) + 1$$

totaal recursief is. Bovendien moet gelden dat dit gelijk is aan

$$\Psi(k, e, j^k(x_1, \cdots, x_k))$$

voor een zekere e. Maar voor die e zou dus ook moeten gelden dat

$$\Psi(k, e, e) = \Psi(k, e, e) + 1$$

Hier constateren we een contradictie en dus bestaat er niet zo'n Ψ voor totaal recursieve functies.

De volgende stelling die we introduceren is de Smn-stelling. We zullen niet op het bewijs ingaan.

Stelling 11
Voor iedere $m \geq 1$ en iedere $n \geq 1$ is er een $m+1$-plaatsige primitief recursieve functie S_n^m zodat voor alle $e, x_1, \cdots, x_m, y_1, \cdots, y_n$ het volgende geldt:

$$\varphi^{(n)}_{S_n^m(e,x_1,\cdots,x_m)}(y_1, \cdots y_n) \simeq \varphi_e^{m+n}(x_1, \cdots x_m, y_1, \cdots, y_n)$$

Gevolg (Recursiestelling)
Er is een primitieve functie H zodat voor iedere e, f, x we hebben dat:

$$\varphi^{(1)}_{H(e,f)}(x) \simeq \varphi_e^{(1)}(\varphi_f^{(i)}(x))$$

Gevolg
Voor iedere partieel recursieve functie $F : \mathbb{N}^k \rightharpoonup \mathbb{N}$ met $k \geq 1$ is er een index e zodat voor alle $x_1, \cdots, x_k$ het volgende geldt:

$$\varphi_e^{(k)}(x_1, \cdots, x_k) \simeq F(x_1, \cdots, x_k, e)$$

Bewijs
Laat f een index van F, dus $\varphi_f^{(k+1)}(x_1, \cdots, x_{k+1}) \simeq F(x_1, \cdots, x_{k+1}, e)$ voor alle $x_1, \cdots, x_{k+1}$. Laat nu g een index die voor alle $h, y, x_1, \cdots, x_k$ voldoet aan:

$$\varphi_g^{(k+2)}(h, y, x_1, \cdots, x_k) \simeq \varphi_h^{(k+1)}(x_1, \cdots, x_k, S_k^1(y, y))$$

Merk op dat de uitdrukking aan de rechterkant van de vergelijking een partieel recursieve functie van $h, y, x_1, \cdots, x_k$ is, dus dat bewijst dat zo'n index

g bestaat. We definiëren nu $e = S^1_k(S^1_{k+1}(g,f), S^1_{k+1}(g,f))$, dan hebben we:
$\varphi^{(k)}_e(x_1, \cdots, x_k) \simeq$
$\varphi^{(k)}_{S^{(1)}_k(S^1_{k+1}(g,f), S^1_{k+1}(g,f))}(x_1, \cdots, x_k) \simeq$ met behulp van de Smn-theorie
$\varphi^{(k+1)}_{S^q_{k+1}(g,f)}(S^1_{k+1}(g,f), x_1, \cdots, x_k) \simeq$
$\varphi^{(k+2)}_g(f, S^1_{k+1}(g,f), x_1, \cdots, x_k) \simeq$ door de keuze van g
$\varphi^{(k+1)}_f(x_1, \cdots, x_k, S^1_k(S^1_{k+1}(g,f), S^1_{k+1}(g,f))) \simeq$ door de definitie van e
$\varphi^{(k+1)}_f(x_1, \cdots, x_k, e) \simeq$ door de keuze van f
$F(x_1, \cdots, x_k, e)$

Definitie 12
Een deelverzameling $A \subset \mathbb{N}^k$ noemen we *recursief opsombaar* als er een primitief recursieve deelverzameling $U \subseteq \mathbb{N}^{k+1}$ bestaat zodat

$$A = \{\vec{x} \in \mathbb{N}^k \mid \exists y.(y, \vec{x}) \in U)\}$$

Dan zullen we als slot van dit hoofdstuk een belangrijke stelling introduceren die ons zal helpen bij het uiteindelijke bewijs.

Stelling 13 (Turing)
De verzameling $\{(e, \vec{x}) \in \mathbb{N}^{k+1} \mid \vec{x} \in dom(\varphi^{(k)}_e)\}$ is recursief opsombaar, maar niet recursief.

3 Het systeem van Peano Arithmetic

3.1 Inleiding

Het systeem van eerste-orde Peano Rekenkunde, ook wel PA zoals we het in het vervolg zullen noemen, is een theorie in de taal $\mathcal{L}_{\mathrm{PA}} = \{0, 1; +, \cdot\}$. Hierbij zijn 0 en 1 constantes en $+$ en $\cdot$ binaire functie symbolen. PA heeft de volgende axioma's:

1. $\forall x \neg(x + 1 = 0)$
2. $\forall xy(x + 1 = y + 1 \rightarrow x = y)$
3. $\forall x(x + 0 = x)$
4. $\forall xy(x + (y + 1) = (x + y) + 1)$
5. $\forall x(x \cdot 0 = 0)$
6. $\forall xy(x \cdot (y + 1) = (x \cdot y) + x)$
7. $\forall \vec{x}[(\varphi(0, \vec{x}) \wedge \forall y(\varphi(y, \vec{x}) \rightarrow \varphi(y + 1, \vec{x}))) \rightarrow \forall y \varphi(y, \vec{x})]$

Punt 7 is bedoeld als een axioma voor iedere formule $\varphi(y, \vec{x})$. Deze axioma's worden inductie axioma's genoemd. Zo'n verzameling van axioma's, die gegeven wordt door een of meerdere algemene symbolen φ die invloed hebben op alle formules, noemen we een axioma schema. In ons geval, hebben we het over het inductie schema.
Uit de bovengenoemde axioma's concluderen we dat PA oneindig veel axioma's heeft; er is namelijk geen eindige $\mathcal{L}_{\mathrm{PA}}$-theorie met dezelfde modellen als PA. Het is duidelijk dat de verzameling $\mathbb{N}$ samen met de elementen 0 en 1 en de standaard optelling en vermenigvuldigin een model van PA is. Dit model noemen we ook wel het standaard model en we noteren het als $\mathcal{N}$. Maar dit is niet het enige model van PA, PA heeft namelijk ook niet standaard modellen.

We definiëren voor iedere $n \in \mathbb{N}$ een term $\overline{n}$ van $\mathcal{L}_{\mathrm{PA}}$ waarvoor door recursie geldt dat: $\overline{0} = 0$ en $\overline{n+1} = \overline{n} + 1$. En dus toegepast: $\overline{5} = (((((0 + 1) + 1) + 1) + 1) + 1)$. Termen die van de vorm $\overline{n}$ zijn worden nummers genoemd. We introduceren nu een nieuwe constante c en we bekijken bij de taal $\mathcal{L}_{\mathrm{PA}} \cup \{c\}$ de volgende verzameling axioma's:

$$\{\text{axioma}'\text{s van PA}\} \cup \{\neg(\mathrm{c} = \overline{\mathrm{n}} \mid \mathrm{n} \in \mathbb{N}\}$$

Uit de Compactheidsstelling volgt dat deze verzameling axioma's een model heeft. Dit is dan een model waarin de interpretatie van de constante c geen

natuurlijk getal is.

De theorie PA is erg sterk; het kan alle recursieve functies representeren en ook het overgrote deel van de elementaire getaltheorie kan door PA worden beschreven. De theorie van PA helpt ons ook bij het uiteindelijke bewijs van de stelling van Church. Omdat we hier uiteindelijk terecht willen komen, moeten we eerst wat uitwerken over de elementaire getaltheorie. We beginnen met de standaard eigenschappen van optellen en vermenigvuldigen:

1. $\text{PA} \vdash \forall x(x = 0 \vee \exists y(x = y + 1)$
2. $\text{PA} \vdash \forall xyz(x + (y + z) = (x + y) + z)$
3. $\text{PA} \vdash \forall xy(x + y = y + x)$
4. $\text{PA} \vdash \forall xyz(x + z = y + z \to x = y)$
5. $\text{PA} \vdash \forall xyz(x \cdot (y \cdot z) = (x \cdot y) \cdot z)$
6. $\text{PA} \vdash \forall xy(x \cdot y = y \cdot x)$
7. $\text{PA} \vdash \forall xyz(x \cdot (y + z) = (x \cdot y) + (x \cdot z))$
8. $\text{PA} \vdash \forall xyz(\neg(z = 0) \wedge x \cdot z = y \cdot z \to x = y)$

Al deze eigenschappen, kunnen we bewijzen met inductie. We zullen dit voor de eerste drie eigenschappen laten zien.

1. Laat $\varphi(x)$ de formule $x = 0 \vee \exists y(x = y + 1)$. Het is hier duidelijk dat $\text{PA} \vdash \varphi(0) \wedge \forall y\varphi(y + 1)$. Dus $\text{PA} \vdash \forall x\varphi(x)$.

2. We maken hier gebruik van inductie op z. We kiezen hiervoor $\varphi(z) = \forall xy(x + (y + z) = (x + y) + z)$. Dan geldt volgens axioma 3, vermeld aan het begin van dit hoofdstuk, dat $\text{PA} \vdash \varphi(0)$ en volgens axioma 4: $\text{PA} \vdash \varphi(z) \to \varphi(z + 1)$, omdat

$$\varphi(z) \vdash (x + (y + z)) + 1 = x + ((y + z) + 1) = x + (y + (z + 1))$$

3. Voor het bewijs van deze eigenschap hebben we een dubbele inductie nodig. Eerst passen we de inductie toe op x. Neem hiervoor $\varphi(x) = \forall y(x + y = y + x)$. $\text{PA} \vdash \varphi(0)$ vanwege wederom axioma 3. Er geldt namelijk $\varphi(0) = \forall y(0 + y = y + 0 = \forall y(y = y)$. Nu moeten we laten zien dat $\varphi(x) \to \varphi(x + 1)$. Dit doen we met behulp van axioma 4. Dit klopt ook omdat:

$$\varphi(x) \vdash (x + y) + 1 = x + (y + 1) = y + (x + 1)$$

We moeten nu ook nog inductie toepassen op y, dit is analoog aan inductie op x en zullen we daarom niet laten zien.

We hebben bij bovengenoemde voorwaarden voor het eerst het symbool $\vdash$ gebruikt, deze zullen we daarom eerst nog definiëren.

Definitie 14
De relatie $\Gamma \vdash \varphi$ betekent dat er en bewijsboom is met natuurlijke deductie, waarbij de veronderstellingen ofwel elementen van Γ zijn, of van de vorm $\forall x(x = x)$ voor een zekere variable x.

We gaan nu verder met de volgende feiten die van belang zijn. Laat $\varphi(x, y)$ de volgende formule zijn: $\exists z(x + (z + 1) = y)$. Dan geldt in PA dat φ een discrete lineaire ordening met kleinste element definieert, waarvoor het principe van het kleinste getal geldt.

1. PA $\vdash \neg\varphi(x, x)$
2. PA $\vdash \varphi(x, y) \wedge \varphi(y, z) \rightarrow \varphi(x, z)$
3. PA $\vdash \varphi(x, y) \vee x = y \vee \varphi(y, x)$
4. PA $\vdash x = 0 \vee \varphi(0, x)$
5. PA $\vdash \varphi(x, y) \rightarrow (y = x + 1 \vee \varphi(x + 1, y))$
6. PA $\vdash \exists w\psi(w) \rightarrow \exists y(\psi(y) \wedge \forall x(\psi(x, y) \rightarrow \neg\psi(x)))$
7. PA $\vdash \varphi(x, x + 1)$

Het schema gegeven bij puntje 5 hierboven, noemen we de 'least number principle', afgekort LNP.
Het principe van het kleinste getal houdt in dat iedere niet-lege set van natuurlijke getallen een kleinste element heeft. Dit kunnen we bewijzen met inductie. Voor LNP is er gebruik gemaakt van [6].

Stelling 15
Iedere niet-lege set van natuurlijke getallen heeft een kleinste element.

Bewijs
We laten A een niet lege deelverzameling van $\mathbb{N}$. We willen laten zien dat A een kleinste element heeft, dus dat er een element $a \in A$ bestaat zodat geldt $a \leq n$ voor alle $n \in \mathbb{N}$. Zoals gezegd gaan we dit laten zien met inductie op de volgende formule: $P(n)$: als $n \in A$ dan heeft A een kleinste element.

- $P(0)$ klopt duidelijk, omdat $0 \leq n$ voor alle $n \in \mathbb{N}$.
- Bij deze inductiestap willen we laten zien dat als $P(0) \wedge P(1) \wedge P(2) \wedge \ldots \wedge P(n)$ geldt, dat dan ook $P(n+1)$ geldt. We nemen daarom aan dat $P(0), P(1), P(2), \ldots, P(n)$ allemaal waar zijn. Daarnaast nemen we ook aan dat $n+1 \in A$. We kunnen nu weer twee gevallen onderscheiden.

  1. $\neg\exists m(m \in A \wedge m < n+1)$
     In dit geval geldt dat $n+1$ het kleinste element in A is.
  2. $\exists m(m \in A \wedge m < n+1)$
     Omdat in dit geval $P(m)$ waar is, heeft A een kleinste element volgens de definitie van P hierboven vermeld.

  In beide gevallen is in ieder geval $P(n+1)$ waar.
- Dus met behulp van inductie hebben we laten zien dat $P(n)$ waar is voor alle $n \in \mathbb{N}$. We weten dat A niet leeg is dus we kunnen een willekeurige n kiezen. Bovendien weten we dat $P(n)$ waar is en dus dat A een kleinste element heeft.

De volgorde waarin we de laatste voorwaarden hebben neergezet is van belang en we introduceren hiervoor zelfs een nieuw symbool. We schrijven vanaf nu: $x < y$ voor $\exists z(x + (z+1) = y)$, $\exists x < y$ voor $\exists x(x < y \wedge \cdots)$ en $\forall x < y$ voor $\forall x(x < y \rightarrow \cdots)$. Verder schrijven we ook nog $x \leq y$ voor $x = y \vee x < y$ en $x \neq y$ voor $\neg(x = y)$. Het maken van deze afkortingen is noodzakelijk als we formele uitspraken willen doen en ook om het vervolg makkelijker te maken.

3.2 Elementaire getaltheorie in PA

Getaltheorie is een tak van de wiskunde die gehele getallen bestudeert. Een oudere term voor getaltheorie is arithmetic. Getaltheorie kan in verschillende groepen worden opgedeeld en een groep hiervan is de elementaire getaltheorie. In de elementaire getaltheorie worden dingen berekend zonder andere takken van de wiskunde toe te passen. Voorbeelden van vraagstukken in de elementaire getaltheorie zijn de grootste deler te vinden en een getal opdelen in priemgetallen. We willen elementaire getaltheorie nu gebruiken in PA en ons beginpunt is de stelling van deelbaarheid met rest. Bij deelbaarheid met rest worden twee gehele getallen door elkaar gedeeld, met als uitkomst een quotiënt en een rest. De stelling luidt als volgt:

Stelling 16
PA $\vdash \forall xy(y \neq 0 \rightarrow \exists ab(x = a \cdot y + b \wedge 0 \leq b < y))$
Dit houdt in dat PA bewijst dat a en b in de stelling uniek gedefinieerd zijn. We laten dit bewijs zien met inductie.

Bewijs
Als $x = 0$ dan geldt $0 = 0 \cdot y + 0$. De stelling klopt dus voor $x = 0$. Als $x = a \cdot y + b \wedge 0 \leq b < y$ dan geldt volgens puntje 5 bij de laatsgenoemde voorwaarden dat $b + 1 < y \vee b + 1 = y$. Als $b + 1 < y$ dan geldt dat $x + 1 = a \cdot y + (b + 1)$ en in het andere geval, dus als $b + 1 = y$ dat $x + 1 = (a + 1) \cdot y + 0$.
Nu moeten we nog laten zien dat het een unieke oplossing is. Dit gaan we doen met behulp van een contradictie. Stel we hebben dat $x = a \cdot y + b = a' \cdot y + b'$ met $0 \leq b, b' < y$. Als we aannemen dat $a < a'$, dan kunnen we zeggen dat $a+1 \leq a'$. Dus als we dit invullen krijgen we: $a' \cdot y \geq (a+1) \cdot y = a \cdot y + y > a \cdot y + b = x$. Hier hebben we dus een tegenspraak. We mogen concluderen dat $a' \leq a$ en vanwege symmetrie dat $a' = a$ en ook $b' = b$.

In de notatie die we bij stelling 12 hebben gegeven, noemen we a het geheeltallige deel van x bij een deling door y en b de rest van x na deling met y. Net zoals in de vorige paragraaf introduceren we weer een verkorte notatie.

$x|y \equiv \exists z(x \cdot z = y)$
$\text{irred}(x) \equiv \forall v \leq x(v|x \rightarrow v = 1 \vee v = x$
$\text{prime}(x) \equiv x > 1 \wedge \forall yz(x|(y \cdot z) \rightarrow x|y \vee x|z)$

Hier staat irred voor irreducibel en prime voor priem.
Verder weten we dat geldt PA $\vdash \forall xy\exists!z((z = 0 \wedge y < x) \vee x = z + y)$. We voegen daarom nu het functiesymbool $-$ aan de taal $\mathcal{L}$ toe, en dit functiesymbool heeft het volgende axioma:

$$\forall xy((x < y \wedge x - y = 0) \vee (x = y + (x - y)))$$

Stelling 17
PA $\vdash \forall x(x > 1 \rightarrow (\text{irred}(x) \leftrightarrow \text{prime}(x)))$

Bewijs
Als $\text{prime}(x)$ geldt en $v|x$ dan geldt dus dat $v \cdot z = x$. Als dit waar is dan geldt $x|v$ dan en slechts dan als $v = x$ of $x|z$ dan en slechts dan als $v = 1$. Dit zijn precies de eigenschappen van irreducibel, dus geldt $\text{irred}(x)$. Nu nemen we het omgekeerde aan; stel we weten $\text{irred}(x)$ en $x > 1$. We nemen voor $P(v)$ de volgende formule: $\forall yz \leq v(y \cdot z \leq v \wedge x|(y \cdot z) \rightarrow x|y \vee x|z)$. We

kunnen laten zien dat $\forall w(\forall v < wP(v) \rightarrow P(w))$, na inductie kunnen we concluderen dat $\forall wP(w)$ en dit duidt op prime(x). Dus als we aannemen dat $\forall v < wP(v)$ en $y, z \leq w$ zodat geldt $y \cdot z \leq w, x|(y \cdot z), x \nmid y$ en $x \nmid z$. Dan geldt dat $y, z > 1$ en als we dan gebruik maken van Stelling 12 mogen we aannemen dat $y < x$, omdat, als dit niet het geval zou zijn, we y zouden kunnen vervangen met de rest bij deling door x. Nog een keer gebruik makend van stelling 12, kunnen we zeggen: $x = a \cdot y + b$ met $0 \leq b < y$. Als $b > 0$ hebben we dat $b \cdot z = (x - a \cdot y) \cdot z = x \cdot z - a \cdot y \cdot z$. Dus we hebben dan dat $x|(b \cdot z), x \nmid b, x \nmid z$ en $b \cdot z < y \cdot z \leq w$. Dit is een contradictie met $\forall v < wP(v)$. Dan nemen we nu aan dat $b = 0$. Met het begrip irreducibel van x geldt dan $y = 1 \vee y = x$. Dit is ook in beide gevallen een contradictie. Dus we kunnen concluderen $P(w)$.

Stelling 18
PA $\vdash \forall x(x > 1 \rightarrow \exists v(prime(v) \wedge v|x))$

Bewijs
Dit bewijs is een stuk minder lang, omdat we hier gebruik kunnen maken van veel dat we al eerder hebben bewezen of andere stellingen. Als $x > 1$, dan weten we dat $x|x$ altijd geldt en dus hebben we $\exists w(w > 1 \wedge w|x)$. Vanwege LNP, weten we dat er zo'n kleinste w bestaat en bovendien weten we dat deze w irreducibel is. Bij stelling 13 hebben we laten zien dat dan dus ook geldt dat prime(w) geldt.

We definiëren nu wederom twee eigenschappen, namelijk x is de macht van het priemgetal v en x is een priemmacht. Deze definities zien er als volgt uit:

pow$(x, v) \equiv x \geq 1 \wedge$ prime$(v) \wedge \forall w \leq x(w > 1 \wedge w|x \rightarrow v|w)$
pp$(x) \equiv \exists v \leq x$ pow(x, v)

Voor prime(v), willen we bepalen voor iedere $y > 0$ zijn v-deel, dit houdt in de hoogste macht van v dat te delen is door y. We noteren dit als $y \uparrow v$. Voorbeelden die het misschien verduidelijken zijn: $16 \uparrow 2 = 4$, $16 \uparrow 3 = 2$, $16 \uparrow 4 = 2$, $16 \uparrow 5 = 1$.
We verwachten als axioma: pow$(y \uparrow v, v) \wedge (y \uparrow v)|y \wedge (y \uparrow v) \cdot v \nmid y$. Voordat we dit axioma mogen gebruiken, moeten we het natuurlijk eerst bewijzen. We willen het volgende bewijzen:

PA $\vdash \forall yv\exists!z((z = 0 \wedge (y = 0 \vee \neg$prime$(v)))\vee$ pow$(z, v) \wedge z|y \wedge z \cdot v \nmid y)$

Als pow(y, v), dan nemen we nu $z = y$. Als we dit niet doen, dan hebben we

namelijk: $\exists w \leq y(w|y \wedge v \nmid w)$ en dus $\exists z \leq y \exists w \leq y(y = w \cdot z \wedge v \nmid w)$ en met behulp van LNP, kunnen we concluderen dat er een kleinste z bestaat. Dan geldt $\text{pow}(z, v)$ en ook $z|y$. Als $z \cdot v|y$ dan geldt $y = w' \cdot x \cdot v = w \cdot z$. Dus geldt $w = w' \cdot v$, maar dit is in tegenstelling tot wat we eerder hadden gezegd, namelijk $v \nmid w$. Dus zo'n z bestaat, bewezen met contradictie.

We gaan gelijk verder met het volgende wat bewezen moet worden.

Lemma 19
$\text{PA} \vdash \forall xy(x|y \leftrightarrow \forall v \leq x(\text{pp}(v) \wedge v|x \rightarrow v|y))$.

Bewijs
Het deel van het bewijs van links naar rechts is triviaal dus zal ik niet laten zien. Hetzelfde geldt voor het bewijs van rechts naar links met de gevallen $y = 0 \vee x = 1$. We gaan de rest van het bewijs laten zien met een contradictie. We nemen $x > 1$ als kleinste element zodat $\exists y \geq 1(\forall v \leq x(\text{pp}(v) \wedge v|x \rightarrow v|y) \wedge x \nmid y)$ geldt en we nemen ook de kleinst mogelijk bijbehorende y. Net zoals bij delen door x met een rest, kunnen we aannemen $y < x$. We nemen daarom ook weer $x = a \cdot y + b$ met $0 \leq b < y$. Als $0 < b$ hebben we een contradictie wat betreft het kleinste geval van y. Dus we kunnen zeggen dat $b = 0$ en $x = a \cdot y$. We bekijken nu het geval met $a > 1$. Dan heeft a een deler v dat een priemgetal is volgens stelling 14, omdat we weten dat geldt $\text{pp}(v), v|x$ en $v|y$. Maar in dit geval hebben we $\text{pp}((y \upharpoonright v) \cdot v) \wedge (y \upharpoonright v) \cdot v|x \wedge (y \upharpoonright v) \cdot v \nmid y$. En dit is een tegenstelling en dus hebben we bewezen wat we wilden.

We kunnen vanaf nu het kleinste gemeenschappelijke veelvoud en de grootste gemeenschappelijke deler van twee getallen definiëren. Daarnaast kunnen we ook hun basisschappen bewijzen in PA. We zullen het kleinste gemeenschappelijke veelvoud in het vervolg noteren als lcm (least common multiple) en de grootste gemeenschappelijke deler als gcm (greatest common divisor).
We nemen $x, y \geq 1$. Omdat geldt dat $x|x \cdot y$ en $y|x \cdot y$, bestaat er een uniek kleinste getal $w > 0$ met $x|w \wedge y|w$. We zullen deze w noteren als $\text{lcm}(x, y)$. Het is duidelijk dat $\text{lcm}(x, y) \leq x \cdot y$.
Als we schrijven $x \cdot y = a \cdot \text{lcm}(x, y) + b, 0 \leq b < \text{lcm}(x, y)$. We zien hier dat $x|b \wedge y|b$ dus als $b \geq 0$ krijgen we een tegenstelling met het feit dat $\text{lcm}(x, y)$ het kleinste element moet zijn. Dus we kunnen nu zeggen $x \cdot y = a \cdot \text{lcm}(x, y)$, voor een unieke a. Deze noemen we $\text{gcd}(x, y)$. Als we zeggen $\text{lcm}(x, y) = y \cdot z$, krijgen we $x \cdot y = \text{gcd}(x, y) \cdot y \cdot z$ dus $x = \text{gcd}(x, y) \cdot z$ en dus $\text{gcd}(x, y)|x$; hetzelfde geldt voor $\text{gcd}(x, y)|y$.

Stelling 20
PA $\vdash \forall xy \geq 1, \exists a \leq y, b \leq x(a \cdot x = b \cdot y + \gcd(x, y)))$

Bewijs
We gaan dit bewijs geven met inductie.

- Voor $x = 1$ kies $a = 1, b = 0$. Dan klopt het voor $x = 1$.
- Voor $x > 1$ laat $y = c \cdot x + d, 0 \leq d < x$. Als we deze vergelijking delen door $\gcd(x, y)$ krijgen we $y' = c \cdot x' + d'$ met $d' < x' \leq x$ en daarnaast $\gcd(x', d') = 1$.
- Volgens de inductiehypothese hebben we dan $u \cdot d' = v \cdot x' + 1$ voor een geschikte u, v. Dus $v \cdot x' = u \cdot d' - 1$. We kwadrateren nu beide kanten, dan krijgen we voor een zekere a' en b': $a' \cdot x' = b' \cdot d' + 1$. De volgende stap die we uitvoeren is vermenigvuldigen met $\gcd(x, y)$. Dit geeft $(a' + b' \cdot x) \cdot x = b' \cdot y + \gcd(x, y)$. Als laatste laten we $(a' + b' \cdot c) = c' \cdot y + a''$ met $0 \leq a'' < y$. Als we dit invullen krijgen we $a'' \cdot x = (b' - c' \cdot x) \cdot y + \gcd(x, y)$. Voor de laatste vergelijking geldt $a'' < y$ en omdat $(b' - c' \cdot x) \cdot y \leq a'' \cdot x < x \cdot y$ hebben we $(b' - c' \cdot x) < x$.
- Dit is precies wat me moesten bewijzen en ds zijn we klaar.

Om goed te kunnen begrijpen wat er hierna gaat komen, gaan we eerst iets algebraïsch laten zien. Stel de ons een aantal getallen is gegeven: $x_0, \cdots x_{n-1}$. We laten $m = \max(x_0, \cdots x_{n-1}, n)!$. Dan geldt voor alle getallen i, j met $0 \leq i < j < n$ we de getallen $m(i + 1) + 1$ en $m(j + 1) + 1$ hebben die beide relatief priem zijn. Dit houdt in dat beide getallen deelbaar zijn door een priemgetal p en dat het verschil $m(j - i)$ ook deelbaar is door p. Omdat p een priemgetal is, zou moeten gelden dan $p|m$ en ook $p|m(i + 1) + 1$ en dit is een tegenspraak. Omdat we zeggen dat $x_i < m(i + 1) + 1$ voor iedere i, kunnen we stellen met behulp van de Chinese resttheorie dat er een getal a bestaat zodat voor iedere i het volgende geldt:

$$a \equiv x_i \ mod \ m(i + 1) + 1$$

We zeggen dan dat het getal a, of liever het paar (a, m), de reeks $x_0, \ldots, x_{n-1}$ codeert in een zin. De volgende stelling die we introduceren, bevat drie essentiële eigenschappen van coderen in PA.

- Voor iedere x, is er een reeks die begint met een x.
- Iedere reeks kan langer gemaakt worden.

- Een technische conditie.

Weer zullen we afkortingen introduceren: $\mathrm{rm}(x, y)$ geeft ons de rest van x bij deling door y en $(a, m)_i$ geeft $\mathrm{rm}(a, m \cdot (i+1)+1)$, dus de rest van a bij deling door $m(i+1)+1$.

Stelling 21

1. $\mathrm{PA} \vdash \forall x \exists a, m((a, m)_0 = x)$
2. $\mathrm{PA} \vdash \forall y x a m \exists b n(\forall i < y((a, m)_i = (b, n)_i) \wedge (b, n)_y = x$
3. $\mathrm{PA} \vdash \forall a m i((a, m)_i \leq a)$

Bewijs

1. We nemen $m = x$ en $a = 2x + 1$, dan geldt

$$rm(a, m \cdot (0+1)+1 = rm(2x+1, x+1) = x$$

2. We bekijken het volgende:

   $\mathrm{PA} \vdash \forall y x a m \exists u(\forall i < y((a, m)_i < u) \wedge x < u \wedge y < u)$
   $\mathrm{PA} \vdash \forall u \exists v \geq 1 \forall i \leq u(i \geq 1 \rightarrow i|v)$
   $\mathrm{PA} \vdash \forall u v(\forall i \leq u(i \geq 1 \rightarrow i|v) \rightarrow \forall i j(0 \leq i < j \leq u \rightarrow \gcd((i+1) \cdot v + 1, (j+1) \cdot v + 1) = 1))$
   De eerste vergelijking kan worden bewezen door inductie op y, de tweede bij inductie op u en de derde met behulp van de eigenschappen van gcd, die we eerder al vermeld hebben.
   Dus bij een gegeven y, x, a, m, kiezen we een u die ervoor zorgt dat de eerste vergelijking klopt, v zodat de tweede vergelijking klopt voor u en we zeggen tot slot $n = v$. We krijgen dan:

$$\forall i < y((a, m)_i < (i+1) \cdot n + 1), \quad x < (y+1) \cdot n + 1$$
$$\forall i j(0 \leq i < j \leq y \rightarrow gcd((i+1) \cdot n + 1, (j+1) \cdot n + 1) = 1$$

   We willen nu een b vinden zodat:

$$(\forall i < y((a, m)_i = (b, n)_i)) \wedge x = (b, n)_y$$

   Om dit te kunnen doen, passen we inductie toe. Stel voor $k < y$ dat er een b' is zodat:

$$(\forall i < k((a, m)_i = (b', n)_i)) \wedge x = (b', n)_y$$

We kunnen nu zien dat voor alle $k < y$ geldt:

$\exists w((y+1)\cdot n+1|w \wedge \forall i < k((i+1)\cdot n+1|w) \wedge gcd(w,(k+1)\cdot n+1) = 1)$

We passen om dit te zien inductie toe op k en de eigenschappen van n. Neem nu een w waarvoor dit geldt. Dan bestaat er volgens stelling 20 een $u \leq (k+1)\cdot n+1)$ zodat

$$rm(u \cdot w, (k+1)\cdot n+1) = 1$$

Zeg nu $b = b'+u\cdot w\cdot(b'\cdot n\cdot(k+1)+(a,m)_k)$. Dan geldt $(b,n)_y = (b',n)_y = x$ vanwege $(y+1)\cdot n+1|w$ en $i < k \rightarrow (b,n)_i = (b',n)_i = (a,m)_i$ vanwege $(i+1)\cdot n+1|w$. Hiermee kunnen we zeggen:

$(b,n)_k = \mathrm{rm}(b,(k+1)\cdot n+1)$
$(b,n)_k = \mathrm{rm}(b'+b'\cdot n\cdot(k+1)+(a,m)_k,(k+1)\cdot n+1)$
$(b,n)_k = \mathrm{rm}(b'\cdot((k+1)\cdot n+1)+(a,m)_k,(k+1)\cdot n+1)$
$(b,n)_k = \mathrm{rm}(a,m)_k$

Dit maakt de laatste inductiestap af, en hiermee zijn we dan ook klaar met het bewijs.

3. Dit bewijs is triviaal en zullen we niet verder behandelen

3.3 Recursieve functies in PA

Definitie 22
Een $\mathcal{L}_{\mathrm{PA}}$-formule φ noemen we een Δ_0-formule als alle kwantoren begrensd zijn in φ, dus van de vorm $\forall x < t$ of $\exists x < t$, voor een zekere term t die de variabele x niet bevat.
Een $\mathcal{L}_{\mathrm{PA}}$-formule φ noemen we een Σ_1-formule als zij van de vorm $\exists y_1 \cdots y_t \psi$ met ψ een Δ_0-formule.
We schrijven dan ook wel: $\varphi \in \Delta_0$ en $\varphi \in \Sigma_1$.

Definitie 23
Laat $A \subseteq \mathbb{N}^k$ een k-plaatsige relatie. Een $\mathcal{L}_{\mathrm{PA}}$-formule $\varphi(x_1,\cdots x_k)$ met k vrije variabelen, representeert A, als voor alle $n_1,\cdots,n_k \in \mathbb{N}$ geldt:

$$(n_1,\cdots,n_k) \in A \Rightarrow PA \vdash \varphi(\overline{n_1},\cdots,\overline{n_k})$$
$$(n_1,\cdots,n_k) \notin A \Rightarrow PA \vdash \neg\varphi(\overline{n_1},\cdots,\overline{n_k})$$

Laat $F : \mathbb{N}^k \rightarrow \mathbb{N}$ een k-plaatsige functie. Een $\mathcal{L}_{\mathrm{PA}}$-formule $\varphi(x_1,\ldots,x_k,z)$ met $k+1$ vrije variabele representeert F als voor alle $n_1,\ldots,n_k \in \mathbb{N}$ geldt:

$$PA \vdash \varphi(\overline{n_1},\ldots,\overline{n_k},\overline{F(n_1,\ldots,n_k)})$$
$$PA \vdash \exists! z\varphi(\overline{n_1},\ldots,\overline{n_k},z)$$

We zeggen dat een functie of een relatie Σ_1-gerepresenteerd is als er een Σ_1-formule is die haar representeert. Later in de scriptie zullen we nog laten zien dat iedere functie die gerepresenteert wordt, zowel recursief als Σ_1-gerepresenteerd moet zijn.

Definitie 24
Een functie $F : \mathbb{N}^k \to \mathbb{N}$ noemen we bewijsbaar recursief in PA als zij gerepresenteerd wordt door een Σ_1-formule $\varphi(x_1, \ldots, x_k, z)$ waarvoor geldt:

$$PA \vdash \forall x_1 \ldots x_k \exists! z \varphi(x_1, \ldots, x_k, z)$$

Stelling 25
Iedere primitief recursieve functie is bewijsbaar recursief in PA.

Bewijs
We gaan dit wederom bewijzen met inductie, ditmaal naar de opbouw van primitief recursieve functies. Het is duidelijk dat de standaardfuncties x_i voor $i = (1, \ldots, k), x + 1$ en 0, bewijsbaar recursief zijn.
Als de functie $F(\vec{x})$ gedefinieerd is door een samenstelling van G en $H_1, \ldots, H_m$, dan ziet F er dus als volgt uit:

$$F(\vec{x}) = G(H_1(\vec{x}, \ldots, H_m(\vec{x})).$$

Als we nu volgens de inductiehypothese aannemen dat de functies $G, H_1, \ldots, H_m$ gerepresenteerd worden door respectievelijk de Σ_1-formules $\psi, \chi_1, \ldots \chi_m$. Dit heeft als gevolgd dat F wordt gerepresenteerd door de volgende formule:

$$\varphi(\vec{x}, z) \equiv \exists z_1 \ldots z_m (\chi_1(\vec{x}, z_1) \wedge \cdots \wedge \chi_m(\vec{x}, z_m) \wedge \psi(z_1, \ldots z_m, z))$$

Boven genoemde functie is equivalent met een zekere Σ_1-formule, namelijk dat PA $\vdash \forall \vec{x} \exists! z \varphi(\vec{x}, z)$ volgt van de bijbehorende eigenschappen voor $\psi, \chi_1, \ldots \chi_m$.
De cruciale stap die nodig is in dit inductiebewijs, is primitief recursief. We maken hier gebruik van stelling 19.
Neem aan dat $F(\vec{x}, y)$ gedefinieerd is door primitieve recursie van G en H. Dus F is als volgt gedefinieerd:

$$F(\vec{x}, 0) = G(\vec{x})$$
$$F(\vec{x}, y + 1) = H(\vec{x}, F(\vec{x}, y), y)$$

Volgens de inductiehypothese worden G en H door $\psi(\vec{x}, z)$ en $\chi(\vec{x}, u, v, w)$ respectievelijk Σ_1-gerepresenteerd. Dit heeft als gevolg dat F gerepresenteerd wordt door de formule $\varphi(\vec{x}, y, u)$. Deze formule is weer als volgt gedefinieerd:

$$\exists am(\psi(\vec{x}, (a, m)_0) \wedge \forall i < y \ \chi(\vec{x}, (a, m)_i, i, (a, m)_{i+1}) \wedge (a, m)_y = u)$$

Deze uitdrukking moeten we duidelijk zien als een afkorting van iets, want een term in de vorm van $(a, m)_i$, is niet gedefinieerd in de taal $\mathcal{L}_{\mathrm{PA}}$. $\psi(\vec{x}, (a, m)_0)$ is een afkorting voor:

$$\exists c, d < a(a = c \cdot (m + 1) + d \wedge 0 \leq d < m + 1 \wedge \psi(\vec{x}, d))$$

Nog steeds geldt dat de formule φ equivalent is met een Σ_1-formule. Het bewijs dat $\mathrm{PA} \vdash \forall \vec{x}, y \exists! u \varphi(\vec{x}, y, u)$ wordt geleverd door inductie in PA op u. Hierbij worden wederom de eigenschappen van stelling 19 gebruikt.

Stelling 26
Iedere totaal recursieve functie wordt Σ_1-gerepresenteerd in PA.

Bewijs
In de standaard recursietheorie, is er een primitief recursief predicaat T en een primitief recursieve functie U zodat er voor iedere k-plaatsige recursieve functie F er een nummer e bestaat zodat geldt:

$$F(n_1, \ldots, n_k) = m \Leftrightarrow \exists y(T(e, n_1, \ldots, n_k, y) \wedge U(y) = m)$$

De verzameling $\{(n_1, \ldots, n_k, y, m) | T(e, n_1, \ldots, n_k, y) \wedge U(y) = m\}$ is primitief recursief, en volgens stelling 20, wordt deze verzameling ook gerepresenteerd door een Σ_1-formule $\varphi(x_1, \ldots, x_k, y, w)$. Dit kunnen we ook schrijven als:

$$\exists z_1 \ldots z_l P(x_1, \ldots, x_k, y, w, z_1, \ldots, z_l)$$

voor een zekere Δ_0-formule P.
Als $R(z, \vec{x}, w)$ de Δ_0-formule $\exists y < z \exists z_l < z \cdots \exists z_l < z P$ is, dan geldt:

$$PA \vdash \exists y w \varphi(\vec{x}, y, w) \leftrightarrow \exists z w R(z, \vec{x}, w)$$

Als laatste laten we $S(z, \vec{x}, w)$ de Δ_0-formule:

$$w < z \wedge R(z, \vec{x}, w) \wedge \forall u < z \neg \exists v < u R(u, \vec{x}, v)$$

Dan geldt dat $\mathrm{PA} \vdash \exists z w R(z, \vec{x}, w) \leftrightarrow \exists! z \exists w S(z, \vec{x}, w)$ volgens LNP.
Ik beweer dat de functie F wordt gerepresenteerd door de Σ_1-formule $\exists z S(z, \vec{x}, w)$.
Eerst moet gelden dat voor $n_1, \ldots, n_k \in \mathbb{N}$

$$PA \vdash \exists z S(z, \overline{n_1}, \ldots, \overline{n_k}, \overline{F(n_1, \ldots, n_k)})$$

een geldige Σ_1-formule is, en dus bewijsbaar is in PA met behulp van Σ_1-compleetheid. Om te laten zien dat

$$PA \vdash \exists! w \exists z S(z, \overline{n_1}, \ldots, \overline{n_k}, w)$$

kies dan $a \in \mathbb{N}$ zodat $S(\overline{a}, \overline{n_1}, \ldots, \overline{n_k}, \overline{F(n_1, \ldots, n_k)})$ waar is. Vanwege de uniciteit van z in S hebben we:

$$PA \vdash \forall zw(S(z, \overline{n_1}, \ldots, \overline{n_k}, w) \rightarrow z = \overline{a} \wedge w < \overline{a})$$

Omdat daarnaast geldt dat PA $\vdash \forall w < \overline{a}(w = \overline{0} \vee \cdots \vee w = \overline{a-1})$ krijgen we:

$$PA \vdash \overline{F(n_1, \ldots, n_k)} < \overline{a} \textbf{ en}$$
$$PA \vdash \neg S(\overline{a}, \overline{n_1}, \ldots, \overline{n_k}, b) \textbf{ voor alle } b < a, b \neq F(n_1, \ldots, n_k)$$

omdat ook geldt $S \in \Delta_0$. Dus PA $\vdash \exists! w \exists z S(z, \overline{n_1}, \ldots, \overline{n_k}, w)$

4 Het Entscheidungsprobleem

Het Entscheidungsprobleem, in het Nederlands het beslissingsprobleem. Het is beacht door Hilbert en Ackermann. Zoals in de inleiding al vermeld staat, is de vraag van het beslissingsprobleem: Is er een algoritme dat iedere gegeven formule in de predikatenlogica beslist?
Alonzo Church was degene die, met behulp van alles wat in de scriptie staat, een antwoord kon geven op dit probleem. Zijn antwoord op deze vraag was negatief, wat inhoudt dat zo'n algoritme dus niet bestaat. Church redeneerde als volgt:
Laat F een primitief recursieve functie die als volgt gedefinieerd is:

$$F(e,x,y) \;=\; \begin{cases} 0 & \text{als } T(1,e,x,y) \\ 1 & \text{anders} \end{cases}$$

waarbij T het Kleene T-predikaat is, beschreven in hoofdstuk 1.4. Er geldt dan dat F bewijsbaar recursief is in PA (volgens stelling 20). Laat nu $\chi(e,x,y,n)$ een Σ_1-formule die F representeert en ook een totale functie in PA representeert. We hebben hierbij gebruik gemaakt van een eindig aantal inductie axioma's. Daarnaast geldt dat ook voor stelling 21, we een eindig aantal inductie axioma's hebben gebruikt. Laat S de deeltheorie van PA bestaande uit al deze inductie axioma's samen met de eerste zes axioma's die bij PA horen. Dan is S een eindige theorie en voor iedere zin ϕ van de taal $\mathcal{L}_{\mathrm{PA}}$, geldt dat ϕ een gevolg van S is dan en slechts dan als de zin $(\bigwedge_{\varphi\in S}\varphi)\to\phi$ geldig is in de predikaat logica.
Dus als we kunnen laten zien dat er geen algoritme bestaat dat voor zo'n ϕ beslist of ϕ nou wel of niet een gevolg is van S, dan hebben we bewezen dat het Entscheidungsprobleem onoplosbaar is.
Voor willekeurige getallen e en x gelden de volgende equivalenties:

$x \in \mathrm{dom}(\varphi_e^{(1)}) \Leftrightarrow$ volgens hoofdstuk 1 van de scriptie
$\exists y$ zodat $F(e,x,y)=0 \Leftrightarrow$ omdat F wordt gerepresenteerd door χ
$N \vdash \exists y\chi(\overline{e},\overline{x},y,0) \Leftrightarrow$ vanwege de definitie van S
$S \vdash \exists y\chi(\overline{e},\overline{x},y,0)$

Dus, ieder algoritme dat beslist of een gegeven zin uit $\mathcal{L}_{\mathrm{PA}}$ wel of niet een gevolg is van S, geeft ons een algoritme dat beslist of x wel of niet in het domein van $\varphi_e^{(1)}$ zit. Maar dit zou betekenen dat laatstgenoemde verzameling recursief is en dat is een contradictie met stelling 12. Hiermee is de stelling van Church bewezen.

5 Bibliografie

[1] Alonzo Church. *An unsolvable problem of elementary number theory.* American Journal of mathematics (1936): pagina 345-363

[2] A.M. Turing. *On computable numbers, with an application to the Entscheidungsproblem.* J. of Math (1936): pagina 230-265

[3] J. van Oosten. *Gödels Incompleteness Theorems*, 2015

[4] J. van Oosten. *Basic Computability Theory*, 2013

[5] William Gasarch. *Primitive Rec, Ackerman's Function, Decidable, Undeciable, and Beyond Exposition.* Geraadpleegd 9 mei 2016. University of Maryland; **http://www.cs.umd.edu/ gasarch/COURSES/452/S14/dec.pdf**

[6] Hanspeter Fischer. *The Well-Ordering Principle.* Geraadpleegd 25 mei 2016. Ball State University;
http://www.cs.bsu.edu/homepages/fischer/math215/wellorder.pdf