HET LUCAS' ARGUMENT:

Norbert van Ettinger (Studentnummer: 0244236)
Dit stuk is als miniscriptie geschreven in het kader van mijn studie wijsbegeerte. Het behandelt het Lucas Argument in het eerste hoofdstuk en geeft een eigen visie hierop in het tweede.

Het Lucas Argument:

I. Elke machine is een instantie van een formeel systeem.
II. Dus gegeven een machine die consistent is en in staat is om eenvoudige rekenkunde te doen, is er volgens de stelling van Gödel- een ware rekenkundige zin die de machine niet zal produceren, namelijk de Gödelzin van het systeem dat hoort bij de machine.
III. Wij kunnen de waarheid van die Gödelzin inzien.
IV. Ergo machines vormen geen adequaat model van de menselijke geest.

Bespreking van het Lucas argument:
Hier zal een globale omschrijving van de verschillende onderdelen van het argument worden gegeven. Voor nadere verklaringen met betrekking tot het argument wordt verwezen naar het artikel van Prof. Albert Visser onder de titel "Lucas' Argument"

ad I : '*Elke machine is een instantie van een formeel systeem.*'
Machine : Onder machine wordt verstaan een computer. De machine is zo geprogrammeerd dat deze een eindeloze reeks waarheden (de axioma's en bewijzen afgeleid uit de axioma's) produceert.
Formeel systeem : wordt gedefinieerd door taal, axioma's en de afleidingsregels. De taal is opgebouwd uit syntactische (grammaticale) regels en Semantische (betekenis) regels. De axioma's zijn niet strijdig met elkaar. De afleidingsregels zijn instructies hoe je beginnende bij de axioma's formele bewijzen kunt construeren. De computer heeft een formeel systeem geï mplementeerd in zijn programma. Het programma produceert een eindeloze reeks waarheden zijnde de geprogrammeerde axioma's of afgeleide bewijzen uit de axioma's.

ad II: '*Dus gegeven een machine die consistent is en in staat is om eenvoudige rekenkunde te doen, is er volgens de stelling van Gödel- een ware rekenkundige zin die de machine niet zal produceren, namelijk de Gödelzin van het systeem dat hoort bij de machine.*'
Consistentie: Uitgaande van een systeem dat de eerste orde logica omvat is er geen zin waarvan bewezen kan worden dat deze zowel waar als onwaar is . (afwezigheid van contradicties)

De Gödelstelling en de Gödelzin: Omdat in dit argument de Gödelzin en de Gödelstelling een (cruciale) centrale rol spelen zal een summiere beschrijving van deze stelling worden gegeven. Maar alvorens dit te doen volgt eerst een kleine historische introductie. In c.a. 300 v. Christus omschreef Euclides in zijn dertiendelige ''Elementen'' het parallellen postulaat. Dit behelst het idee dat je gegeven het platte vlak maar één parallelle lijn kan trekken door een punt naast een andere lijn. Dit, alsmede het overige werk van Euclides bleef 2000 jaar lang onaangetast, de standaard in de wiskunde. Niemand die ooit een inconsistentie (een bewering die niet logisch uit de gegeven veronderstellingen volgt) in het werk heeft kunnen vinden. Er werd gedacht dat de Euclidische meetkunde de meetkunde van de natuur was. Totdat George Friedrich Bernard Rieman met een strijdig axioma kwam en stelde dat alle evenwijdige lijnen elkaar in het oneindige snijden. Uit dit met de Euclidische meetkundige strijdige axioma bleek evenzogoed een consistente meetkunde te kunnen worden afgeleid welke ronduit in tegenspraak bleek te zijn met de even zo consistente Euclidische meetkunde. Uit de angst dat deze 'Crisis' in de meetkunde over zou slaan op andere delen van de Rekenkunde deden logici als Frege hun best de aritmetica te ondersteunen en proberen de gehele wiskunde op een rigoureuze wijze op te bouwen door deze op zo weinig mogelijk axioma's te laten grondvesten (deduceren). Feitelijk was het Fundament van de verzamelingenleer welke Frege gebruikte, gebaseerd op slechts twee Axioma's welke reeds eerder door Cantor waren geformuleerd. Frege's Fundamentele werk '' Die grundlagen der Arthmetik'' lag klaar naast de persen toen Russel met een tegenstrijdigheid in de verzamelingenleer kwam welke bekend is geworden als de Russel Paradox. R = {A| A is een verzameling en $A \notin A$ } Voor de Verzameling van Verzamelingen die niet een element zijn van zichzelf geldt zowel $R \notin R$ als zowel $R \in R$. (R is de universele Russel verzameling). Een gepopulariseerde versie van Russels paradox is de barbier van Sevilla die alleen mannen scheert die zichzelf niet scheren. Scheert die zichzelf? Het probleem in de Verzamelingen leer van Cantor en Frege was een axioma dat de Russel verzameling toestond te bestaan. Om deze nieuwe crisis het hoofd te bieden gingen wiskundigen als Hilbert, Russel en Whitehead aan de slag om net als Frege de wiskunde opnieuw te grondvesten maar waarin de Russel- en aanverwante paradoxen en andere contradicties niet konden ontstaan. Ze wilden een ''principia Mathematica'' neerleggen naar analogie van Newtons ''Principia mechanica''. Ze waren na 20 jaar reeds drie delen gevorderd toen Gödel met schokkende nieuwe stellingen kwam waar hij de bewijzen ook nog voor leverde. Gödel's eerste onvolledigheidstelling zegt dat geen enkel complex wiskunde systeem volledig was. Welk axioma je ook kiest, je kunt altijd betekenisvolle wiskundige beweringen doen waarvan de juistheid of de onjuistheid binnen het systeem niet aangetoond kan worden. De tweede onvolledigheidstelling is nog vernietigender. Hij toonde aan dat het onmogelijk was te bewijzen dat een gegeven complex wiskunde systeem consistent was.

Globale uiteenzetting van Gödel's onvolledigheidstelling:
De Gödelstelling wordt bewezen met behulp van een variant op de leugenaars paradox. De leugenaars paradox luidt: ''Deze zin is niet waar!''. Deze paradox wordt nagebootst in een eerste orde rekenkundige logische taal b.v. Peano. Hoe gaan we dat doen? Bedenk dat we een bewijsregime hebben voor een eerste orde logische taal 'FOL'. De bewijzen hierin zijn opgebouwd uit een eindig aantal regels bestaande uit symbolen. Stel nu dat we een coderingssysteem hebben dat de symbolen omzet in enen en nullen. Deze enen en nullen kunnen weer gedacht worden weergaven te zijn van getallen. Dus kunnen we de natuurlijke getallen gebruiken om er elke regel van symbolen in weer te geven. Op deze wijze zijn beweringen over volgordes van symbolen omgezet in analoge beweringen over natuurlijke getallen. Zo kunnen we dan ook beweringen over bewijzen weergeven in beweringen over natuurlijke getallen. Stel nu dat onze FOL zelf een rekenkundige taal is (b.v. Peano). Dan kunnen we deze rekenkundige taal ook representeren op overeenkomstige wijze in Natuurlijke getallen (dus de zelfde rekenkundige taal). Dit geeft ons dan de mogelijkheid beweringen in deze taal te coderen, zodat de gecodeerde beweringen uitspraken doen over de bewijsbaarheid in de taal. Populair gezegd, de taal bewijst dat het bewijst. Gödel liet zien hoe dit allemaal moest en liet vervolgens zien dat er een specifieke zin G in deze taal is te maken die van zichzelf zegt dat ze niet bewijsbaar is in het deductie systeem waarmee we zijn gestart.
We kunnen ons afvragen of deze bewering in G waar is of niet. Als het niet waar is dan moeten we dat ook kunnen bewijzen. Maar het bewijsregime is hier duidelijk over. In de opzet van het Bewijssysteem hebben we gegarandeerd dat alleen ware beweringen bewezen kunnen worden. Dus als G onwaar is dan is het waar en hebben we een bewering die zowel waar als onwaar is, wat onmogelijk is. Dus G moet waar zijn, maar G beweert over zichzelf dat ze niet bewijsbaar is dus dat moet dan waar zijn. Met andere woorden, Godel's Theorema vertelt ons; begin met een degelijk bewijs systeem voor de rekenkunde en vervolgens is daar een ware zin G die niet binnen het systeem te bewijzen is.

De genoemde Gödelzin G zou kunnen luiden : "*Deze zin is niet bewijsbaar in de rekenkunde*."
Deze zin moet naar de rekenkunde vertaald worden. Hierbij stuiten we op twee problemen. Het woordje ''*deze*'' is een zelfrefererend woord en Het begrip ''*bewijsbaarheid* '' is geen rekenkundig begrip maar een element uit de syntaxis (grammatica). Zelfreferentie valt op te lossen met een truc van Quine ; ''Geeft iets leuks wanneer je het achter zijn eigen aanhaling zet.''geeft iets leuks als je het achter zijn eigen aanhaling zet. Zo'n zin zegt iets over zichzelf. Hoe je met het begrip ''bewijsbaarheid in de taal'' omgaat is in de vorige alinea verklaard. Voor dit alles is een vertaling van syntaxis naar rekenkunde nodig. Hoe nu Syntaxis te vertalen naar de rekenkunde?

We nemen Peano als voorbeeld. Peano is dus een eerste orde logische rekentaal. Hoe kunnen we de Gödelzin nu vertalen in een taal die alleen maar over getallen gaat? Dit gaan we doen door aan elke

zin een uniek getal toe te voegen behorende bij die zin. Zinnen en bewijzen worden dus gecodeerd als getallen. Via deze codering kunnen we dan Peano laten ''praten'' over de syntaxis.
Formules zijn symbolen in een eindige taal. Zo ook voor Peano. We kennen in de taal ; ∀ ''alle'',∃''er is'',Λ''en'', S ''de opvolgerfunktie'',0 ''het getal nul'',+ ''plus'',• ''maal''en = ''de identiteit''. S0 in peano is het getal 1 (de opvolger van 0) en SS0 is het getal 2 (de opvolger van de opvolger van 0).

Axioma's zijn :
1. $Sx \neq 0$
2. $Sx = Sy$
3. $x + 0 = x$
4. $x + Sy = S(x+y)$
5. $x \bullet 0 = 0$
6. $x \bullet Sy = x \bullet y + x$
7. $A0 \land \forall x(Ax \rightarrow ASx) \rightarrow \forall x Ax$

Ter illustratie; Voor de optelling van 2 en 3 = 5 in Peano luidt: SS0 + SSS0 = SSSSS0 .
We zien dat er in Peano in het resultaat van de optelling sprake is van het achter elkaar zetten (concatenatie) van opvolgerfunkties van de op te tellen delen.

Als voorbeeld, hoe kunnen we nu de rekenkundige wereld laten praten over deze operatie van concatenatie. Het antwoord luidt, door deze te coderen . Hoe ziet dan de spiegeling van deze concatenatie eruit in de rekenkundige wereld?

Deze vertaling wordt alsvolgt gedaan. Veronderstel reeksen met alleen de karakters 'A' en 'B'. Voorbeelden van reeksen zijn A, AA , B , BA, ABBAAB etc.
Elke unieke reeks krijgt een unieke code in de Numerieke taal overeenkomstig de volgende tabel:

Syntaxis	**Positie stelsel**	**Codering in rekenkunde**
		0
A	$1 \bullet 2^0$	1
B	$2 \bullet 2^0$	2
A A	$1 \bullet 2^1 + 1 \bullet 2^0$	3
A B	$1 \bullet 2^1 + 2 \bullet 2^0$	4
B A	$2 \bullet 2^1 + 1 \bullet 2^0$	5
B B	$2 \bullet 2^1 + 2 \bullet 2^0$	6
A A A	$1 \bullet 2^2 + 1 \bullet 2^1 + 1 \bullet 2^0$	7
A A B	$1 \bullet 2^2 + 1 \bullet 2^1 + 2 \bullet 2^0$	8
A B A	$1 \bullet 2^2 + 2 \bullet 2^1 + 1 \bullet 2^0$	9
A B B	$1 \bullet 2^2 + 2 \bullet 2^1 + 2 \bullet 2^0$	10
B A A	$2 \bullet 2^2 + 1 \bullet 2^1 + 1 \bullet 2^0$	11

We ordenen naar lengte en binnen de lengte naar alfabet.
Bij de codering wordt een positiestelsel toegepast waarin A de waarde 1 heeft en B de waarde 2. In een positiestelsel bepaald de plaats van het cijfer het eraan toegekende gewicht.
Reeksen worden vertaald door de numerieke code vast te stellen van de concatenatie van alle deelreeksen in die reeks teneinde zo elke unieke reeks te vertalen naar een unieke numerieke code. Voor het gemak noemen we de reeks een zin en de deelreeksen woorden in de zin.
Het Algoritme voor de vertaling luidt dan: Als *m* de numerieke representatie van het eerste woord is , *n* de numerieke vertaling van het tweede woord en *c* het aantal karakters van het tweede woord in de weergave van de elementaire syntaxis dan vinden we voor de numerieke representatie van de contaminatie van het eerste en het tweede woord :

1) $m \otimes n = m \bullet (\text{grootste } 2^x \leq (n+1)) + n$

Of uitgedrukt in c, het aantal karakters van het tweede woord in elementaire syntaxis

2) $m\otimes n = m\bullet 2^{c} + n$

(Zie voor een afleiding van deze formules appendix Aop blz. 6)

Een zin is een reeks van woorden achterelkaar. Een Voorbeeld van een eenvoudige zin zou kunnen zijn : ''B AA'' : B*AA = m• (kleinste $2^x > n$) + n = 2 * (grootste $2^x \leq (2+1)$) + 3) = $2\bullet 2^2 + 3 = 11$. of gerekend met het aantal karakters van AA is ; $m\otimes n = 2\otimes 3 = 2\bullet 2^2 + 3 = 11$ en dit is de numerieke code van BAA zijnde de contaminatie van B en AA genoteerd als B*AA . Zo krijgt dus elke specifieke zin zijn eigen numerieke code. Met deze conversie van de rekenkundige FOL zijn we in staat om ook weer te rekenen in de rekenkundige FOL en het daarmee een uitspraak te laten doen over zijn eigen bewijzen. Vervolgens kunnen we dan een rekenkundige zin produceren die hetzelfde effect heeft als de Gödelzin en kan de Gödelstelling voor de rekenkundige FOL worden bewezen. Voor de geï nteresseerde lezer zie lit. 6 . De werkelijke coderingstabel is natuurlijk veel groter dan de hier gepresenteerde en omvat alle karakters welke worden gebruikt in de eerste orde logische (reken) taal.

Evaluatie van het Lucas' argument.

'Elke machine is een instantie van een formeel systeem'.

Aangenomen wordt in het lucas argument dat de computer een programma zal bevatten dat één (star en constant) formeel systeem representeert en bewijzen aflevert binnen het model van dit formele consistente systeem. De Gödel zin kan dan overeenkomstig de onvolledigheidstelling niet door de computer gegenereerd worden omdat de computer alleen ware bewijsbare stellingen produceert. Vanuit het perspectief dat de computer op de in het argument omschreven wijze te werk gaat en alleen maar op deze wijze kan werken lijkt het argument (nog even de vaagheden in sommige onderdelen van de aannames daargelaten) mij juist. Echter het volgende:
Computers (machines uit Lucas argument) bestaan uit hard- en software. Het in het lucas argument omschreven consistente formele systeem zal voor het belangrijkste deel geï mplementeerd zijn in de software, de programmatuur. Welnu, Het Lucas argument veronderstelt dat alleen formele systemen ingeprogrammeerd kunnen worden immers de eerste aanname zegt *''elke machine is een instantie van een formeel systeem*''. Maar we kunnen ook computers bouwen met neurale netwerken met zelflerende programmatuur of biochips toepassen, dus sterk op het menselijke brein gelijkende structuren. Zijn dat ook consistente formele systemen?

"Wij kunnen de waarheid van die Gödelzin inzien."

Laten we nu dan vervolgens de verstandelijke vermogens van de mens als uitgangspunt nemen. Wij kunnen dankzij Gödel wel de juistheid van de Gödel zin inzien. Mogen we daaruit afleiden dat wij niet volgens een consistent model of een formeel systeem denken en begrijpen. Of is het dat wij overstappen op een ander consistent model om de Gödelzin uit het ene model te begrijpen of is het nog weer anders. Mocht het zo zijn dat wij overschakelen op een rijker model om de Gödelzin van een armer model te doorzien dan is er voor dit rijkere model ook weer een Gödelzin. Aangenomen dat we nog een tandje hoger gaan dan lukt het om ook nu weer de Gödelzin in te zien van het voorgaande. Maar uiteindelijk komt er toch een eind aan de rijkheid van het door ons gehanteerde model. en dan is daar toch die Gödelzin. Maar hoe kunnen we die nog herkennen. zijn we daar gegeven de onvolledigheidstelling dan niet blind voor. Zo beschouwd als we maar ver genoeg gaan heeft de mens ook zijn eigen Godelzin en valt het onderscheid met de machine op dit onderdeel helemaal weg en kan het lucas argument vanwege verkeerde aannames op de helling.

Lucas geeft in zijn argument de computer een handicap mee door deze niet de gelegenheid te geven van formeel systeem te wijzigen. Wat is er op tegen om ook de computer een zelfde methode toe te laten passen door regelmatig uitstapjes te laten maken naar een ander consistent model te maken om vandaaruit het te beschouwen model te laten (onder)zoeken op/naar de Gödelzin in het beschouwde model.

In feite wordt getracht twee subjecten met elkaar te vergelijken. De kennis over de computer een door onszelf geschapen object is veel groter en waarschijnlijk redelijk goed maar de kennis met betrekking tot het subject mens en met name de werking van de hersenen is nog lang niet compleet.

De vraag waarom wij de Gödelzin kunnen begrijpen past binnen het onderzoek naar de werking van onze hersenen. Het is maar de vraag of de filosofie hier een antwoord op kan geven. Mochten we weten hoe onze hersenen dit Gödelprobleem en nog vele andere zaken oplossen dan kunnen we vervolgens proberen om dit zelfde mechanisme te implementeren in een computer (het is dan ook maar de vraag of computers dan nog gelijkgesteld mogen worden met formele systemen). Komen we hierbij tot de conclusie dat deze implementatie (vooralsnog) niet mogelijk is dan zou ik de stelling dat mensen geen machines zijn (in ieder geval tot aan een volgende toetsing) wel willen handhaven. maar dan niet op grond van het Lucas'argument maar op grond van het ervaringsgegeven dat de computer tot op heden iets ontbeert wat de mens wel heeft; n.l. bewustzijn.

Diverse overwegingen:

- In het Lucas'argument zitten te veel vaagheden met betrekking tot de aannames

- Als alles met zo weinig onderzoek zou zijn af te leiden uit een simpele redenering waren we dan niet veel verder geweest dan we nu zijn?.

- Als alles in de wereld (mensen even buiten beschouwing gelaten) Gedetermineerd lijken te zijn door natuurwetten. en al deze dingen zijn opgebouwd uit dezelfde bouwstenen (atomen) als de mens hoe is het dan denkbaar dat de mens iets extra's heeft dat niet latent opgesloten ligt in deze elementaire bouwstenen. Is er dan iets buiten deze bouwstenen waar wij geen weet van hebben behalve dan dat, dat iets zich openbaart in de mens. Dit is een vraag waar al lange tijd over wordt nagedacht en die door verschillende stromingen in de filosofie op verschillende wijze wordt beantwoord. Existentialisten geloven in een eigenschappen zoals de wil in de mens die daar aan ontsnappen. Deterministen menen dat ook de mens volledig wordt bepaald door de natuurwetten Maar sedert de laatste jaren begint ook in de natuurkunde het besef te ontstaan dat onvoorspelbaarheid en onzekerheid een rol spelen in de stoffelijke wereld. Waarmee de these van het determinisme in het gedrang komt.

- Afwezigheid van bewijs is nog geen bewijs van afwezigheid. Zoals platlanders levende in een tweedimensionale wereld in een driedimensionaal universum absoluut geen weet hebben van een andere tweedimensionale wereld evenwijdig aan die van hen. Wat staat ons nog te wachten op informatica gebied en wat op het gebied van het onderzoek naar het menselijke brein?

- Tegenover een versimpeld argument als dat van Thomas, staat lijnrecht de mening van veel computer experts met jarenlange ervaring in de informatica. Bill Joy (de geestelijke vader van de computertaal Java) Amerikaans computerdeskundige schrijft in het april nummer van het tijdschrift Wire: " Bij het ontwerpen van programmatuur en microprocessoren heb ik nooit het gevoel gehad dat ik een intelligente machine zat te bedenken. De hard- en software is zo primitief en 'denk'-vermogen is bij de machine zo totaal afwezig, dat ik zoiets als op zijn minst heel verre toekomstmuziek heb beschouwd. Thans echter, nu rekenvermogen op het niveau van menselijke hersenen binnen 30 jaar in zicht lijk te komen dringt zich een nieuwe gedachte op: dat ik wellicht nu bezig ben het gereedschap te maken waarmee technologieën zullen worden gebouwd die de plaats van de mens zullen gaan innemen. Al tientallen jaren voorspelt de wet van Moore correct het exponentiele ontwikkelingstempo van de halfgeleidertechnologie. Elke 18 maanden verdubbelt het rekenvermogen van computers."

- Ook wil ik de krachtmeting enkele jaren geleden tussen IBM Big Blue en Wereldkampioen schaak grootmeester Kasparov in herinnering roepen. Een titanen strijd die in het voordeel van de computer werd beslecht.

Epiloog:
Tegen al deze achtergrond valt toch niet te ontkennen dat er een duidelijke opmars van het kunnen van de computer in de richting van menselijk verstandelijk vermogen zichtbaar is en zou het naï ef zijn het versimplificerende argument van Lucas als een geldig negatief antwoord te beschouwen op de vraag of computers de menselijke geest kunnen imiteren benaderen evenaren of zelfs overklassen .
In dat opzicht lijkt het dat Lucas hetzelfde kunststuk van Gödel van 80 jaar geleden nogmaals wil herhalen. Waar Gödel terecht een bom legde onder de zoektocht van wiskundigen naar een waterdicht consistent model zonder contradicties en paradoxen wil Lucas dezelfde bom nogmaals plaatsen maar nu onder het streven in de ontwikkeling van steeds slimmere computersystemen met inherente kunstmatige intelligentie. Aan de lezer nu te bepalen of hij dat terecht doet en of hij in die opzet is geslaagd.

Appendix A:

Afleiding van codering formules :

1) $m \otimes n = (a_{mp} \bullet 2^{p} + ... + a_{m0} \bullet 2^{0}) \otimes (a_{nq} \bullet 2^{q} + ... + a_{n0} \bullet 2^{0)} \Leftrightarrow$

2) $m \otimes n = (a_{mp} \bullet 2^{p+q+1} + ... + a_{m0} \bullet 2^{q+1}) + (a_{nq} \bullet 2^{q} + ... + a_{n0} \bullet 2^{0)} \Leftrightarrow$

3) $m \otimes n = 2^{q+1} \bullet m + n \Leftrightarrow m \otimes n = 2^{c} \bullet m + n \Leftrightarrow m \otimes n = m \bullet 2^{c} + n$

4) waarin $(1 \bullet 2^{q} + ... + 1 \bullet 2^{0}) \leq n \leq (2 \bullet 2^{q} + ... + 2 \bullet 2^{0}) \Leftrightarrow$

5) $(1 \bullet 2^{q} + ... + 1 \bullet 2^{0)} \leq n \leq (2 \bullet 2^{q+1} + ... + 2 \bullet 2^{1}) \Leftrightarrow (2^{(q+1)} - 1) \leq n \leq (2^{((q+1)+1)} - 1) - 1) \Leftrightarrow$

6) $2^{(q+1)} \leq (n+1) \leq (2^{((q+1)+1)} - 1)$

Vergelijking 6 levert met vergelijking 3 de formulering:

7) $m \otimes n = (grootste 2^{x} \leq (n+1)) \bullet m + n \Leftrightarrow m \otimes n = m \bullet (grootste 2^{x} < (n+1)) + n$

Literatuur :

1. Hoffman, Paul , *De man die van getallen hield. Het verhaal van Paul Erdös en de zoektocht naar de waarheid in de wiskunde*.Uitgeverij Bert Bakker Amsterdam.ISBN 9789035 120471
2. Russel, Bertrand. *Geschiedenis van de westerse filosofie*. Servire/Kosmos –Z&K Uitgevers ISBN 90 215 89966
3. Bill Joy, *Waarom de toekomst ons niet nodig heeft.* NRC handelsblad 26 augustus 2001.
4. Barwise Jon & Etchemendy John, *The language of first order logic*, CSLI , ISBN 0937073903
5. Achterhuis, Hans, *Frankenstein revisited. NRC* magazine M.
6. Nagel, E & Newman, J.R. De stelling van Gödl, Aula paperback 136, 1986.