

De lusvinder genept — een logisch recept

een eenvoudig bewijs van de onbeslisbaarheid van het stopprobleem

Geen programma kan zeggen of een ander ooit klaar is.

Dat beweer ik, en meer: ik laat zien dat het waar is.

Hoe je ook sputtert, ik toon aan dat het klopt:

Je kunt niet voorspellen of een programma ooit stopt.

Stel dat P, een programma, dat tóch kan voorspellen,
en dus broncodes checkt om aan jou te vertellen
of die code in een lus raakt en maar draait en maar spint,
en P schrijft 'dit stopt' als hij géén lussen vindt.

Je geeft P jouw code, met invoer erbij.
En P leest en rekt, en checkt allebei,
en gaat na of dit alles wel stopt na een tijd
en niet door blijft draaien, analyse ten spijt.

Nu, het zit zo: zo'n P kan niet bestaan.
Want stel, jij schrijft hem en komt er mee aan.
Dan kan ik hem gebruiken voor een logische val
zo vilein dat je brein erop breekt met een knal.

De truc die ik uithaal vertel ik je nu.
Ik maak een recept, en ik noem het ding Q.
Q neemt een programma en gaat na of dat stopt
met P kan dat immers, dus je snapt dat dit klopt.

En dit is niet alles. Als P zegt 'een lus'
dan zal Q het beamen, en Q stopt daarna dus.
Maar als P zegt 'dit stopt' gaat Q eindeloos voort
met dat steeds maar herhalen in een lus die ontspoot.

En dit programma, dat ik Q heb gedoopt
gaan we gebruiken, zoals je vast had gehoopt.
Als Q opstart en vraagt om zijn invoer? Zeg: Q.
Q leest dan zichzelf, en . . . Wat krijgen we nu?

Als Q in een lus raakt hebben we brokken.
P zegt dan 'een lus', want P zal niet jokken.
Maar dan moet Q dit beamen en stoppen.
Nu, je begrijpt, dit programma zal floppen.

Maar als Q juist mooi stopt na een tijd
dan zal P dit zien, en vort met de geit
Q moet dan een lus in, want dat was het recept.
Je ziet, ook dit kan niet, dus ik heb je genept.

Wat P ook doet, 't is niet goed of het deugt niet.
Want P zegt hoe het staat en Q doet dat teniet.
Als P eerlijk wil spreken dan liegt hij je wat voor
En zo P wil liegen: het klopt toch door en door.

Het recept dat je hier zag is bekend aan veel koks
uit de logische keuken, en het heet 'paradox.'
Toen je aannam dat die P kon bestaan
kreeg ik je te pakken, en toen ging je eraan.

Hoe kun je ontsnappen aan dit logisch debacle?
Geloof niet in die P, want dat was het obstakel.
Van de tegenspraak die ik liet zien kun je leren:
dat zo'n P kan bestaan kan niemand beweren.

De moraal: niemand kan jou een methode vertellen
die het op 'tilt' gaan van je PC kan voorspellen.
Word niet boos en blij vrolijk, is wat ik je brom,
en zoek zelf naar de fouten, want computers zijn dom.

Jan van Eijck (zeer vrij naar *Scooping the Loop Snooper* van Geoffrey
Pullum, in *Mathematics Magazine*, oktober 2000)