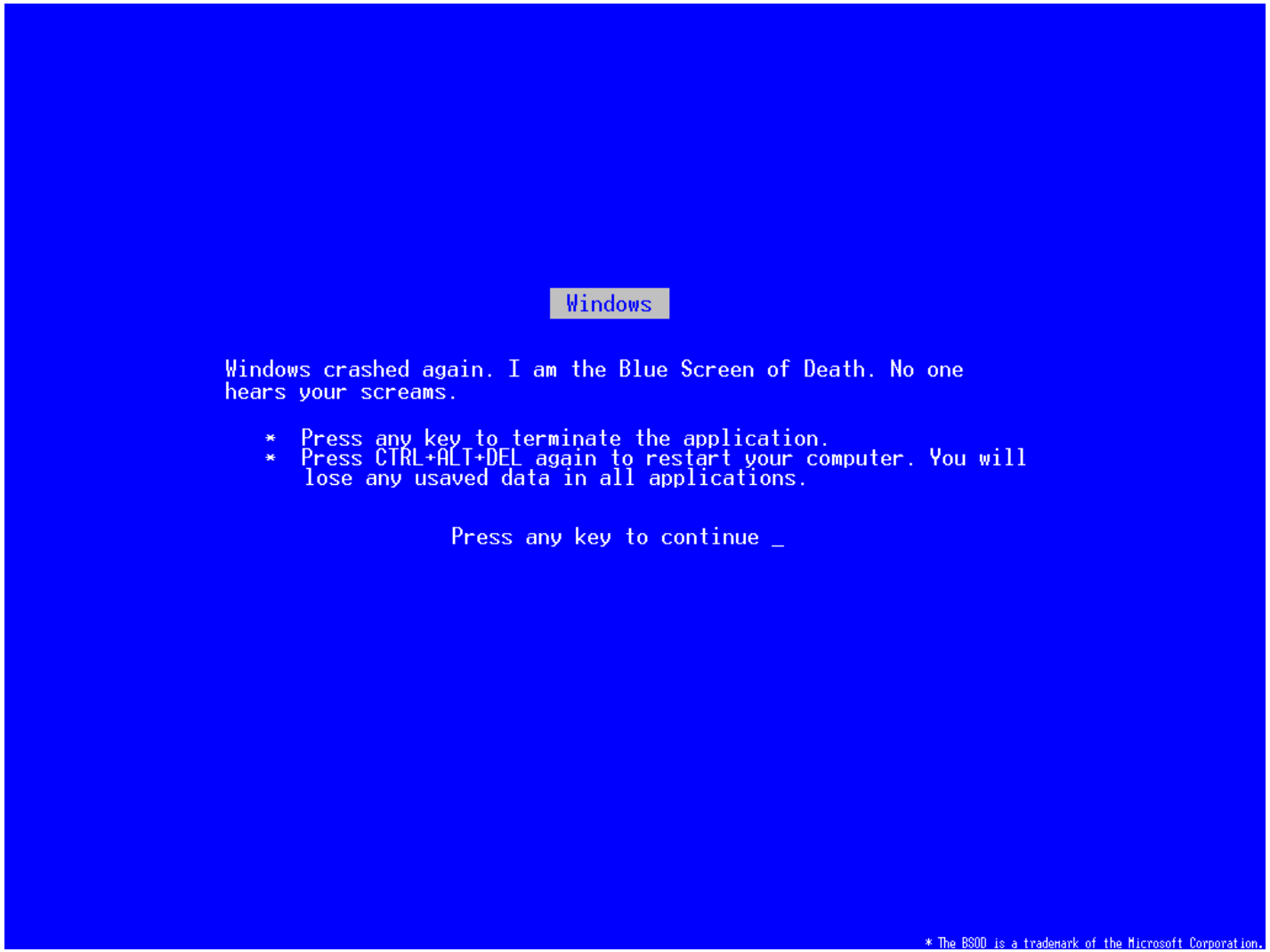
# Computer-ondersteund redeneren: de boekhouder steunt de denker

Herman Geuvers

9 maart 2007

## Computers maken fouten


Windows
Windows crashed again. I am the Blue Screen of Death. No one hears your screams.
* Press any key to terminate the application.
* Press CTRL+ALT+DEL again to restart your computer. You will lose any usaved data in all applications.
Press any key to continue _
* The BSOD is a trademark of the Microsoft Corporation.

## Computers maken fouten

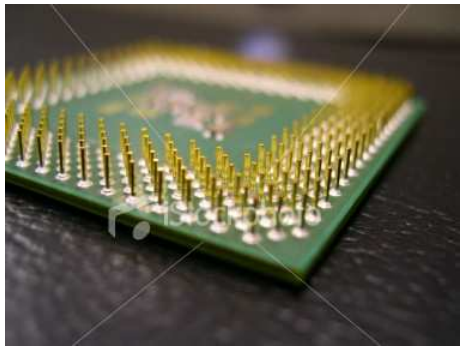
### Ariane

- Ariane 5 raket, 4 juni 1996
- Conversie van 64-bit floating point naar 16-bit integer
- 500 miljoen dollar, 7 miljard ontwikkelkosten.

## Computers maken fouten

### Pentium chip

Intel 1994

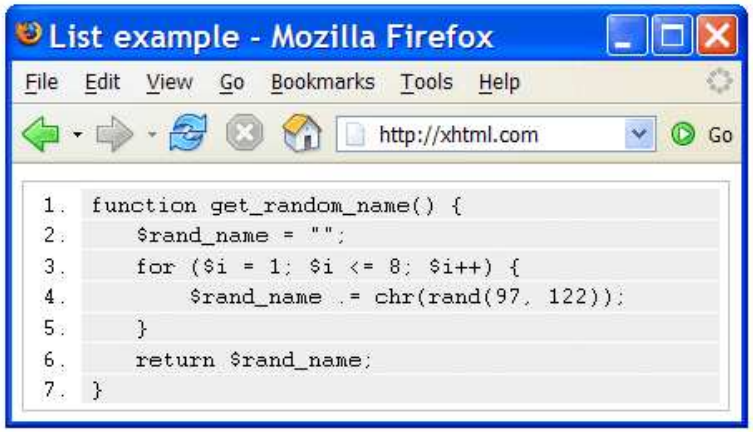


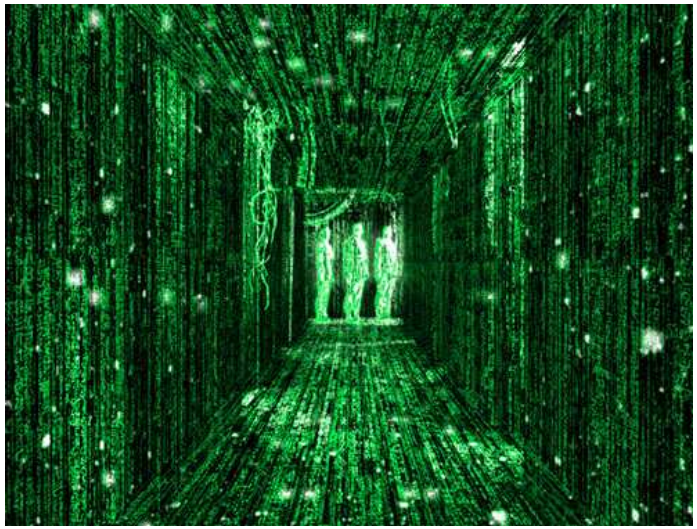
$$4195835.0/3145727.0 = 1.333820449136241000 \text{ (Correcte waarde)}$$

$$4195835.0/3145727.0 = 1.333739068902037589 \text{ (Pentium waarde)}$$

## Hoe voorkomen we fouten?

Gedegen ontwikkelproces



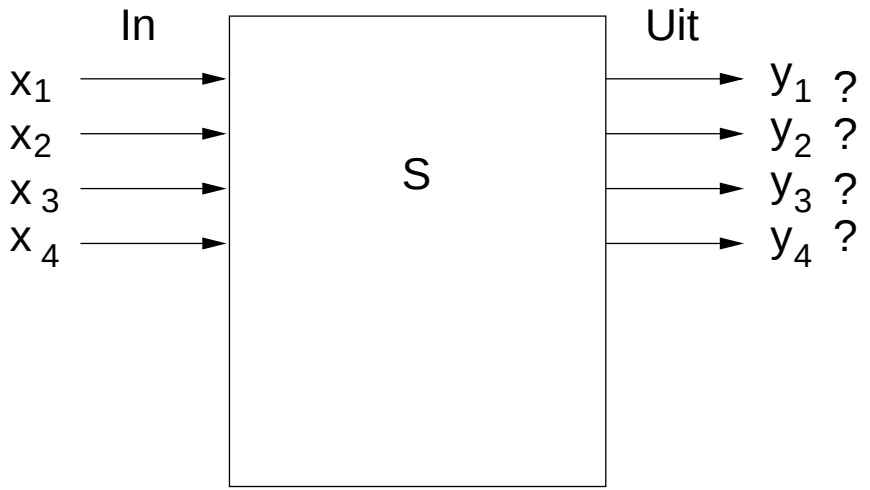



Computer code moet het foutloos doen binnen een bepaalde omgeving

## Hoe voorkomen we fouten?

Eigenschappen verifiëren

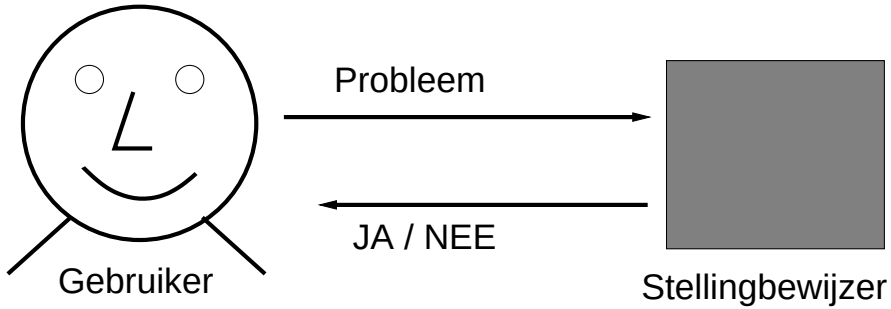
- Testen: Klopt de uitvoer bij de invoer?



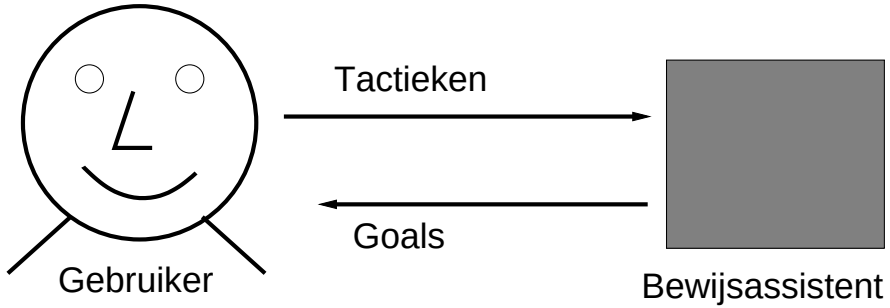

- Correctheidsbewijs

## Bewijsassistent

- Stellingbewijzer? Automatisch?




- Bewijsassistent: Interactief!

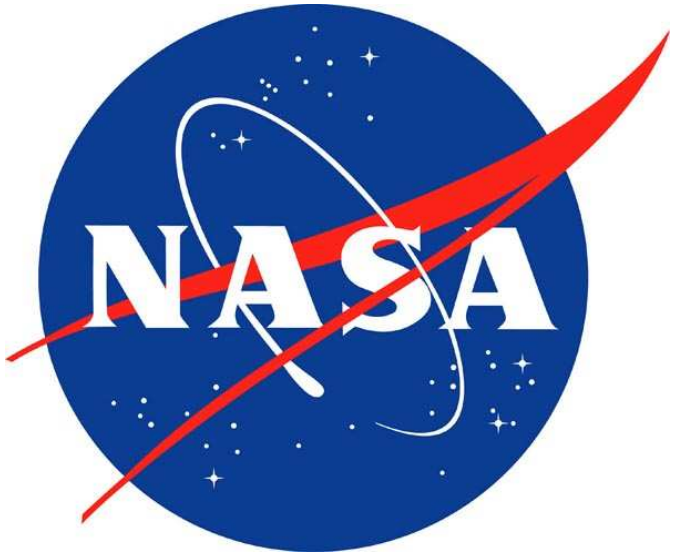


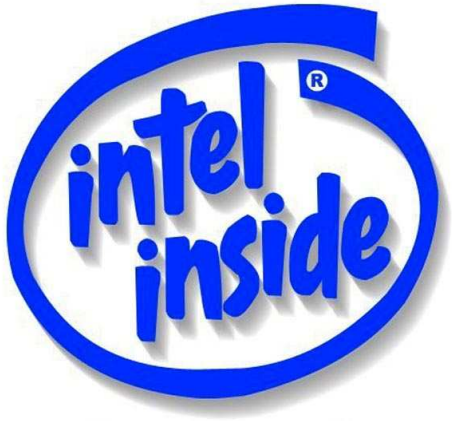
## Holy Grail

> 'Things like even software verification, this has been the Holy Grail of computer science for many decades but now in some very key areas, for example, driver verification we're building tools that can do actual proof about the software and how it works in order to guarantee the reliability.'

Bill Gates, 18 april 2002

## Bewijsassistenten voor software verificatie



## Bewijzen op de computer

- Heel precies, formeel opgeschreven
- Bewijschecken

## Wat is een bewijs?

- Van Dale:

  > Een feit of redenering waaruit de juistheid van een bewering onweerlegbaar blijkt

- wiskundig bewijs: absoluut
- wiskundig bewijs: te reduceren tot heel kleine stapjes. Boekhouderswerk!

## Een bewijs speelt twee rollen

A Een bewijs overtuigt de lezer van de correctheid van het gestelde.

B Een bewijs legt uit waarom het gestelde geldt.

Computer kan rol A overnemen: bewijschecker.

De Bruijn: Automath (eind 60-er jaren)
Eerste bewijscheckers
Nu: bewijsassistenten

## QED manifest

QED is the very tentative title of a project to build a computer system that effectively represents all important mathematical knowledge and techniques.

The QED system will conform to the highest standards of mathematical rigor, including the use of strict formality in the internal representation of knowledge and the use of mechanical methods to check proofs of the correctness of all entries in the system.

## QED manifest

Motivatie

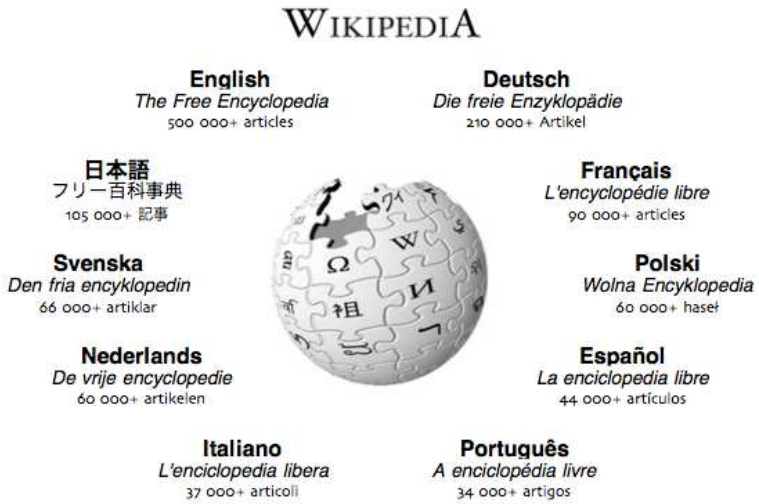
1. De omvang van de wiskunde
2. Het QED systeem als component voor het modelleren, ontwikkelen en verifiëren van software/hardware systemen.

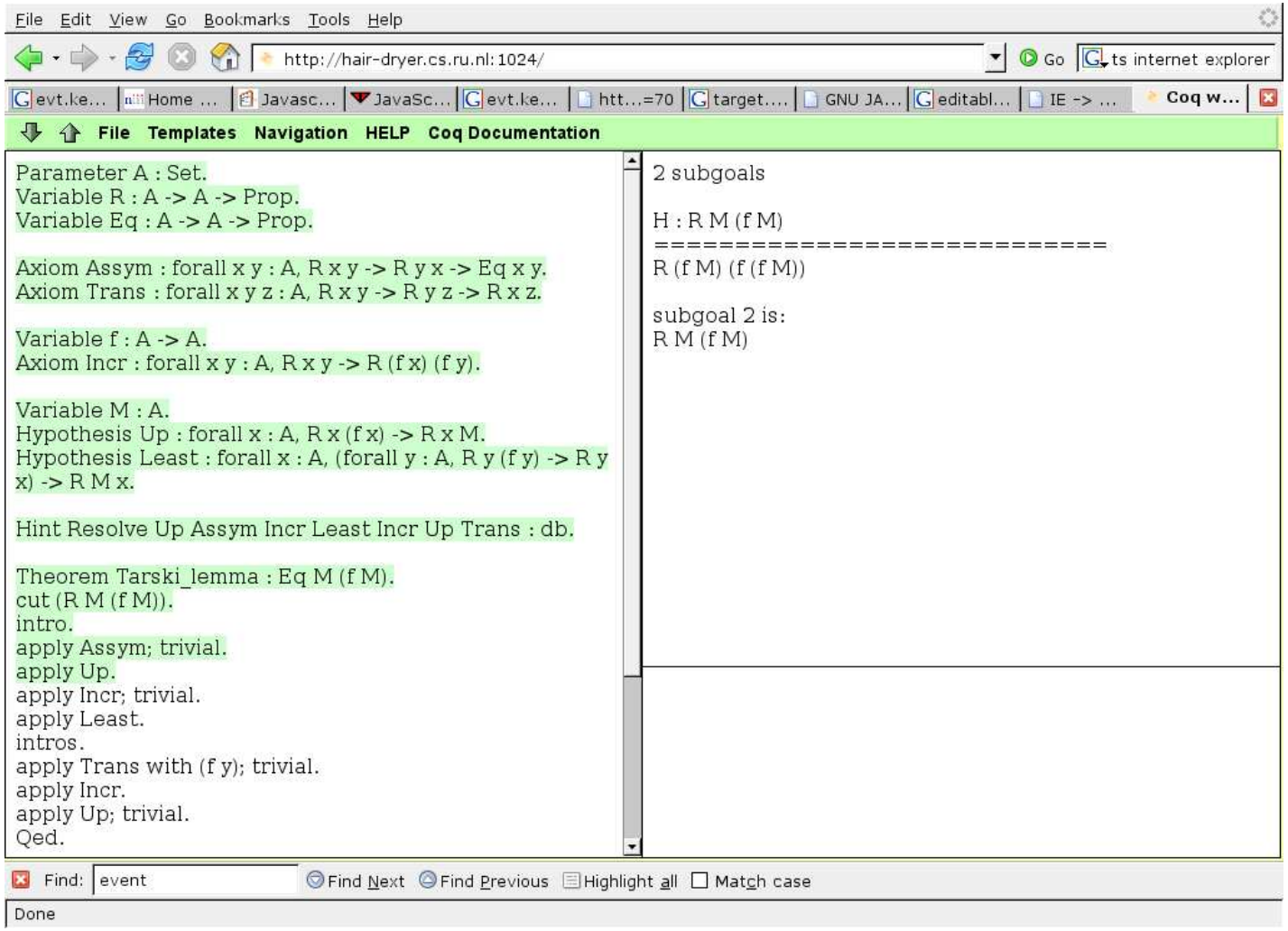
## Te ambitieus?

- Alle wiskunde formaliseren is onrealistisch
  Standaard curriculum van een wiskunde studie formaliseren kost 140 manjaar.
- Huidige bewijsassistenten zijn niet goed genoeg.

## Ambitieus maar niet onmogelijk

- QED manifest: Top down benadering
  Eerste vaststellen wat te doen, in welke volgorde, ...
- Wikipedia: Bottom up benadering
  Lichtgewicht basistechnologie, iedereen kan bijdragen.




File Edit View Go Bookmarks Tools Help
http://hair-dryer.cs.ru.nl:1024/
Go
ts internet explorer
File Templates Navigation HELP Coq Documentation
Parameter A : Set.
Variable R : A -> A -> Prop.
Variable Eq : A -> A -> Prop.
Axiom Assym : forall x y : A, R x y -> R y x -> Eq x y.
Axiom Trans : forall x y z : A, R x y -> R y z -> R x z.
Variable f : A -> A.
Axiom Incr : forall x y : A, R x y -> R (f x) (f y).
Variable M : A.
Hypothesis Up : forall x : A, R x (f x) -> R x M.
Hypothesis Least : forall x : A, (forall y : A, R y (f y) -> R y x) -> R M x.
Hint Resolve Up Assym Incr Least Incr Up Trans : db.
Theorem Tarski_lemma : Eq M (f M).
cut (R M (f M)).
intro.
apply Assym; trivial.
apply Up.
apply Incr; trivial.
apply Least.
intros.
apply Trans with (f y); trivial.
apply Incr.
apply Up; trivial.
Qed.
2 subgoals
H : R M (f M)
============================
R (f M) (f (f M))
subgoal 2 is:
R M (f M)
Find: event
Find Next
Find Previous
Highlight all
Match case
Done

## Bewijsverificatie

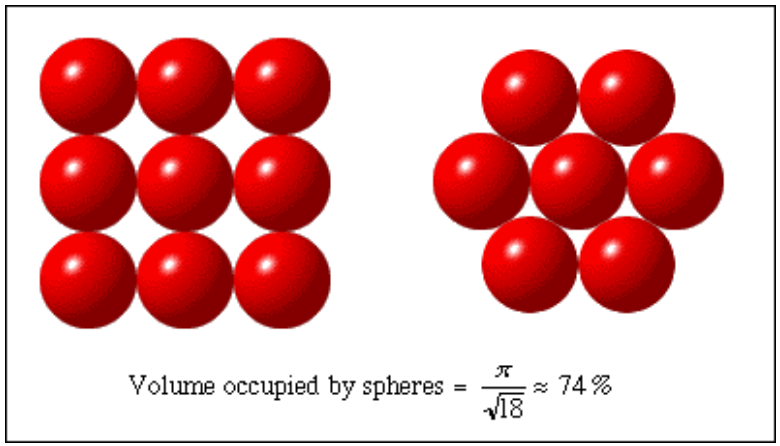
- Geen bovengrens aan de verhouding

$$\frac{\text{lengte van het kortste bewijs van } A}{\text{lengte van } A}$$

- Er zijn echte wiskundige bewijzen die zo groot zijn dat ze niet door één mens gecheckt kunnen worden

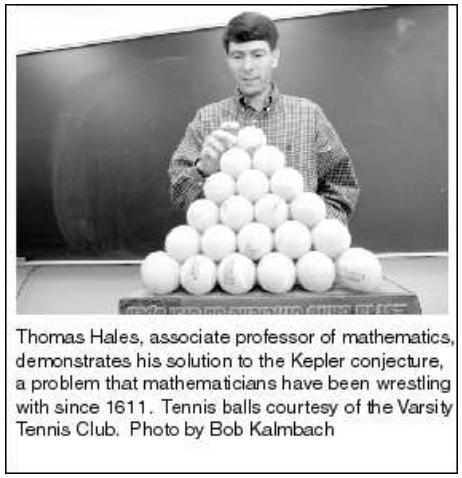
## Vermoeden van Kepler (1611)

De compactste manier om bollen van dezelfde grootte te stapelen, is een piramide.



## Vermoeden van Kepler 1611

- Hales 1998: bewijs met behulp van computer programma's (300 pagina's)



Thomas Hales, associate professor of mathematics, demonstrates his solution to the Kepler conjecture, a problem that mathematicians have been wrestling with since 1611. Tennis balls courtesy of the Varsity Tennis Club. Photo by Bob Kalmbach

- Annals of Mathematics: 99% correct

## Hales' bewijs van het vermoeden van Kepler

Probleem reduceren tot 1039 ongelijkheden van de vorm

$$\frac{\begin{array}{c} -x_1x_3 - x_2x_4 + x_1x_5 + x_3x_6 - x_5x_6 + \\ x_2(-x_2 + x_1 + x_3 - x_4 + x_5 + x_6) \end{array}}{\sqrt{4x_2 \left( \begin{array}{c} x_2x_4(-x_2 + x_1 + x_3 - x_4 + x_5 + x_6) + \\ x_1x_5(x_2 - x_1 + x_3 + x_4 - x_5 + x_6) + \\ x_3x_6(x_2 + x_1 - x_3 + x_4 + x_5 - x_6) \\ -x_1x_3x_4 - x_2x_3x_5 - x_2x_1x_6 - x_4x_5x_6 \end{array} \right)}} < \tan(\frac{\pi}{2} - 0.74)$$

Computer programma's om na te gaan dat deze ongelijkheden gelden.

## Flyspeck project

- Hales: bewijs van het vermoeden van Kepler formaliseren met behulp van bewijsassistenten
- Bewijsassistenten die gebruikt worden: HOL light, Isabelle, Coq

## Software en Hardware correctheid

Systemen moeten aan speciale eisen voldoen

Computers en computer programma's moeten ook aan speciale eisen voldoen



Is de situatie anders voor software en hardware?

## Computer systemen zijn anders

- Discrete systemen: bijna goed bestaat niet
- Hackers: actief zoeken naar (kleine) fouten
- Snelle verspreiding van software en hardware
  Snelle verspreiding van informatie over fouten.

## Formele methoden

Logische en wiskundige methoden en technieken om informatica fenomenen te modelleren, ontwerpen en verfiëren.

- Eigenschappen op een abstracte manier uitdrukken en bestuderen
- Formele methoden ondersteunen door middel van 'tools': computerprogramma's waarmee we de methoden kunnen laten werken.
- Bewijsassistenten zijn zeer generieke tools voor formele methoden.

## Mijn onderzoek

Computer ondersteund redeneren

- Grondslagen
- Toepassen op formaliseren van wiskunde
- Toepassen bij informatica verificatieproblemen

## Grondslagen

- $\lambda$-calculus

$$\lambda x.x + x$$

- termherschrijven en pattern matching

$$\begin{array}{l}\mathsf{rev}\ l := \mathsf{match}\ l\ \mathsf{with} \\ \qquad\qquad \mathsf{nil} \quad \Rightarrow \quad \mathsf{nil} \\ \qquad a :: l' \quad \Rightarrow \quad \mathsf{rev}\ l' +\!\!+\ a\end{array}$$

- logica

$$\dfrac{\dfrac{\forall x.P(x) \to Q(x) \qquad \forall y.P(y)}{P(z) \to Q(z) \qquad P(z)}}{\dfrac{Q(z)}{\forall z.Q(z)}}$$

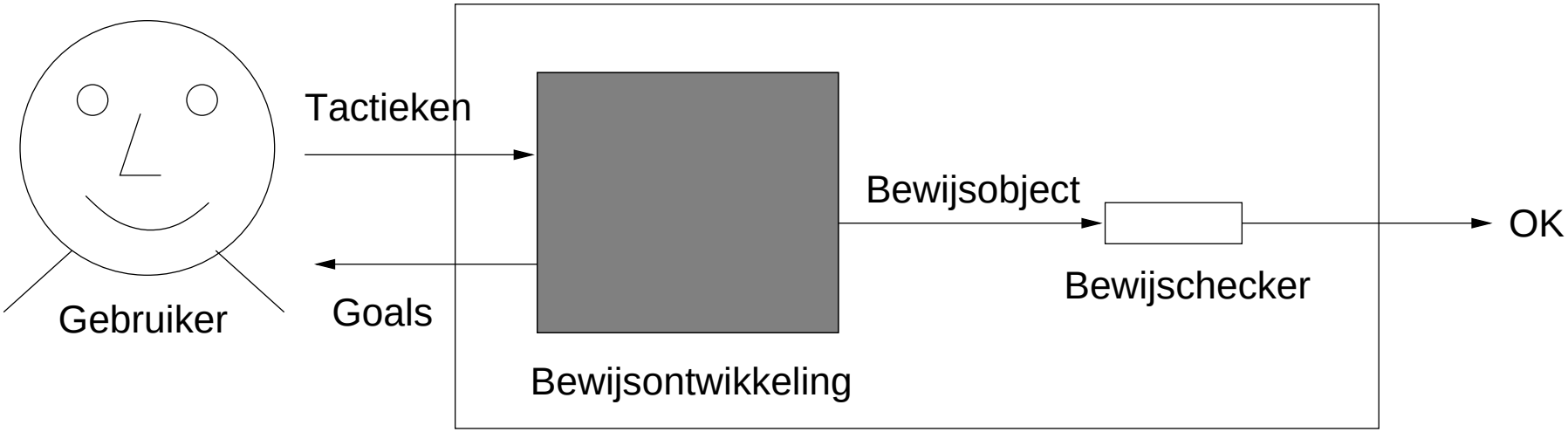
## Typetheorie

- Syntactische notie van "verzameling"
- Formules-als-types isomorfisme

  Een formule is het type van zijn bewijzen

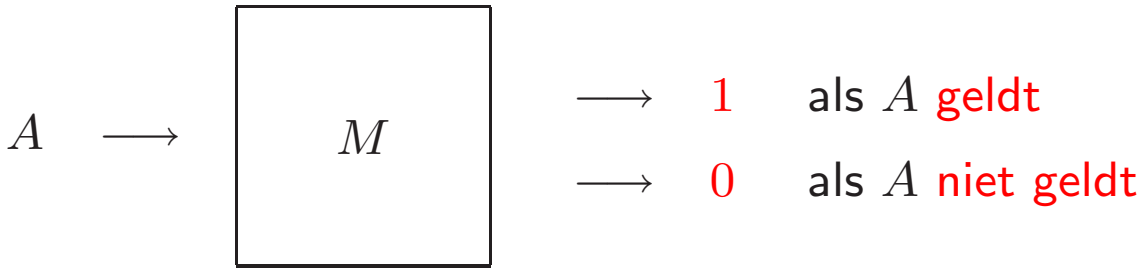
  Bewijzen zijn "gewone" termen
- Bewijs checken = type checken

## Bewijsassistent met bewijsobjecten



## Bewijsassistenten

Een computer programma dat beslist of een formule waar is?




Turing en Church (1936): dat is onmogelijk

## Invoertaal van een bewijsassistent

- Procedureel: je zegt wat de bewijsassistent moet doen.
- Declaratief: je zegt je waar de bewijsassistent heen moet gaan.

## Procedureel versus Declaratief

### Procedureel

vertrek in oostelijke richting

| | |
|---|---|
| na 65 m. | links afslaan |
| na 120 m. | links afslaan |
| na 1430 m. | rechts afslaan |
| na 1640 m. | links afslaan |

etcetera

### Declaratief

| | |
|---|---|
| op Comeniuslaan | ga naar Erasmuslaan |
| op Erasmuslaan | ga naar St. Annastraat |
| op St. Annastraat | ga naar Grootstalselaan |
| op Grootstalselaan | ga naar Hatertseweg |

etcetera

## Procedurele versus Declaratieve formele bewijzen

Procedureel

```
Theorem double_div2: forall (n : nat),  div2 (double n) = n.
simple induction n; auto with arith.
intros n0 H.
rewrite double_S; pattern n0 at 2; rewrite <- H; simpl; auto.
Qed.
```

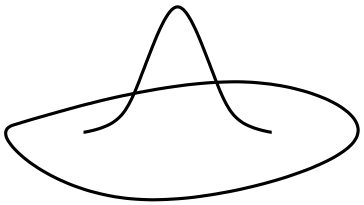
## Procedurele versus Declaratieve formele bewijzen

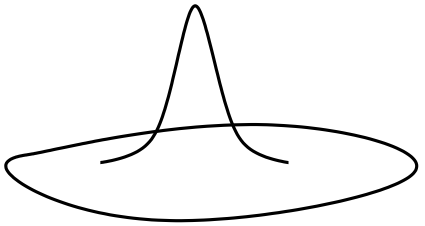
Declaratief

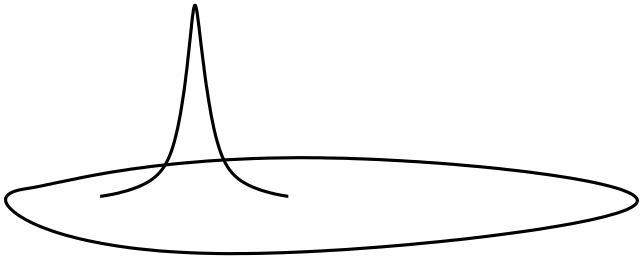
```
Theorem double_div2: forall (n : nat), div2 (double n) = n.
proof.
  assume n:nat.
  per induction on n.
    suppose it is 0.
      thus thesis.
    suppose it is (S m) and Hrec:thesis for m.
      have (div2 (double (S m))= div2 (S (S (double m)))).
           ~= (S (div2 (double m))).
      thus ~= (S m) by Hrec.
  end induction.
end proof.
Qed.
```

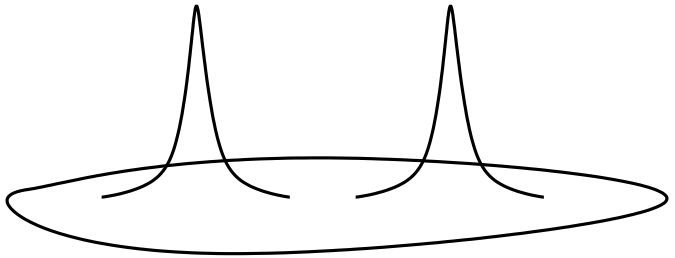
## Formele bibliotheken

Mexicaanse Hoed









## Documentatie

Onze eigen bibliotheek van geformaliseerde wiskunde (CoRN):

- 962 definities
- 3554 lemma's
- totaal 394 pagina's

Literate proving, vergelijkbaar met literate programming

## Interactieve wiskundige documenten

Combineren van document editing en formalizatie

# 1 Nested Intervals

We first give some general constructions and lemmas for nested intervals that will be used in the proof of the Intermediate Value Theorem later.

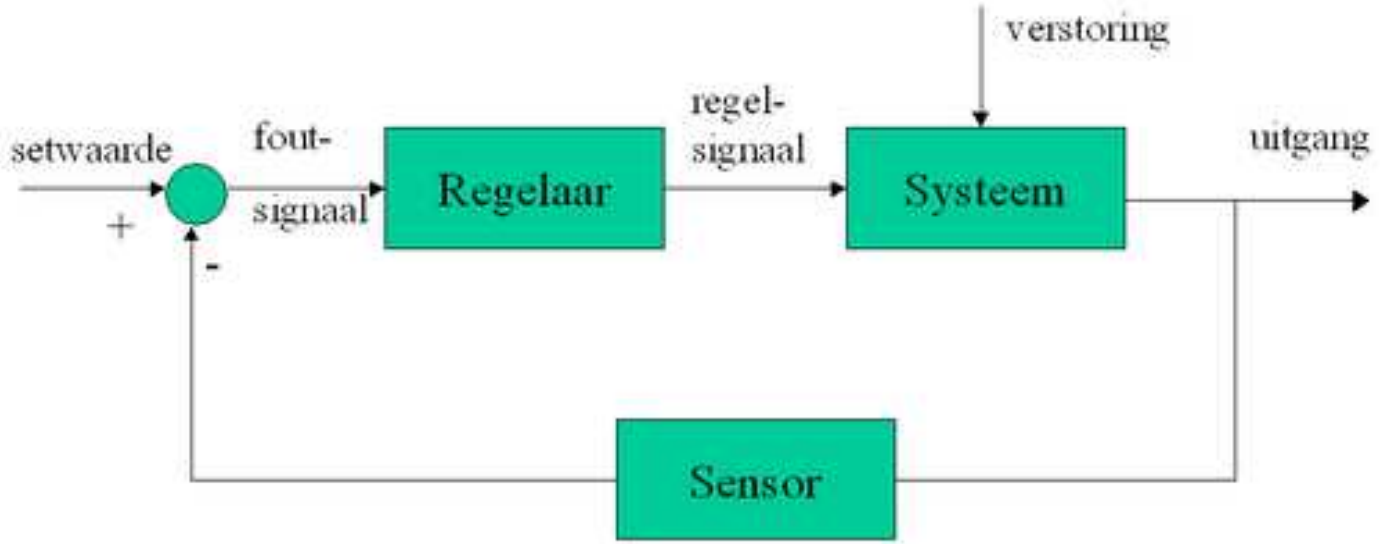
- **Variable 1.** $a, b : \mathbb{N} \to \mathbb{R}$
- **Hypothesis 2.** *$a$ is increasing, i.e. $\forall i \in \mathbb{N}(a_i \leq a_{i+1})$; $b$ is decreasing i.e. $\forall i \in \mathbb{N}(b_i \geq b_{i+1})$; $a$ is below $b$, i.e. $\forall i : \mathbb{N}(a_i < b_i)$; $a$ and $b$ get arbitrarily close, i.e. for every positive real number $\epsilon$, there is an $i$ such that $b_i < a_i + \epsilon$*
- **Lemma 3.** *$a$ is monotone, i.e. $\forall i, j \in \mathbb{N}(i \leq j \to a_i \leq a_j)$*

```
< Lemma a_mon' : forall i j : nat, i<=j -> a i [<=] a j.

  1 subgoal

    a : nat -> IR
    b : nat -> IR
    a_mon : forall i : nat, a i[<=]a (S i)
    b_mon : forall i : nat, b (S i)[<=]b i
    a_b : forall i : nat, a i[<]b i
    b_a : forall eps : IR, Zero[<]eps -> {i : nat | b i[<=]a i[+]eps}
    ============================
     forall i j : nat, i <= j -> a i[<=]a j
```

  - Proof: Trivial using lemma 3

- **Lemma 4.** *$b$ is monotone, i.e. $\forall i, j \in \mathbb{N}(i \leq j \to b_i \leq b_j)$*

## Hybride systemen




- Continue componenten: sensor, klok, thermometer, snelheidsmeter
- Discrete componenten: gaskraan met 3 standen, klok-reset
- Omgeving modelleren
- Abstraheren van de oneindige toestandsruimte

## Onderwijs

- Studenten en docenten: wees actief en interactief
- Het gebruik van computers en computer tools is een actieve bezigheid

## Universitair Onderwijs

- Wiskundige en logische methoden
- Abstract denken leer je alleen op de universiteit
- Leren presenteren = leren inhoud te verwerken en presenteren

## Universitair Onderwijs

Zijn de studenten van nu luier of dommer dan 20 jaar geleden?

NEE absoluut niet.

> 'Students get more excited by stuff that works than by stuff one has to think about.'

Niets is zo practisch als een goede theorie

## Dankwoord

Ik heb gezegd