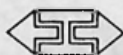


I/O

VIVAT

Orgaan van de
studievereniging Informatica

INTER-ACTIEF



jaargang 2, nummer 2

ALS JE ALS BIJNA AFGESTUDEERDE ELEKTRONICUS NOG NIET HEBT
GESPROKEN MET AT&T EN PHILIPS TELECOMMUNICATIE: 06-0727762

GEMISTE
KANNS

I/O VIVAT

2e jaargang, nummer 2

Orgaan van de
studievereniging
informatica Interactief.

IW-gebouw A 231
Technische Hogeschool
Twente
Postbus 217
7500 AE Enschede.

Redactie.

Johan de Boer
Marin de Nie
Hanke Niestern
Peter Schnitger
Clazien Wezeman

Medewerkers.

Anco Smit
Marcel van Veelen

Kopy.

In de kopijbus bij het
medelingenbord in IW,
of sturen naar de
redactie van I/O Vivat,
IW A 231, THT, Postbus 217,
7500 AE Enschede.

Tel I/A kamer: 3756
Tel hoofdredacteur:
053-895105

Het lidmaatschap van
Interactief kost per
studiejaar 15 gulden.

Advertenties.

P. Schnitger,
Campuslaan 67-215,
Tel. 053-895105.

REDACTIONEEL

Zo, het tweede nummer is lekker snel na het eerste afgerond. Als we zo doorgaan halen we het negende nummer nog wel.

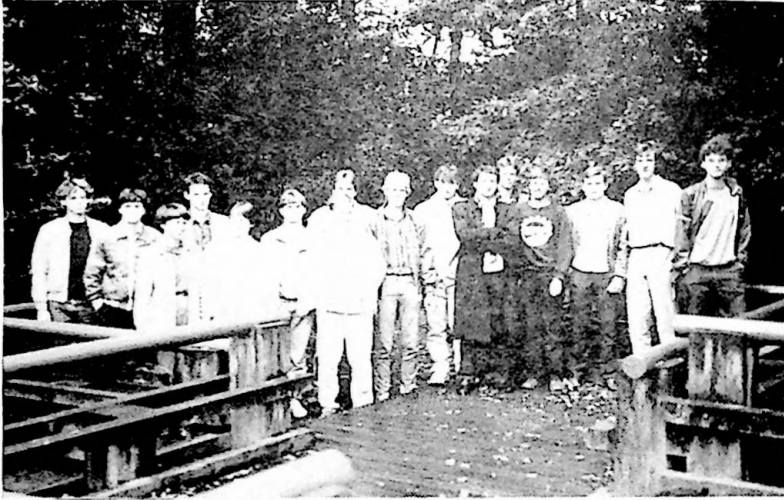
In dit nummer twee stukken door IMF-medewerkers geschreven, misschien aanleiding voor andere medewerkers om ook eens wat te schrijven?

Het hoofdredacteur-schap is met ingang van de tweede jaargang gesplitst. Clazien is verantwoordelijk voor de inhoud en Peter voor de financiën.

INHOUD

Interactievelingen	2
Boekencommissie	2
Van het bestuur	3
A readers introduction to Twintel	4
Grauwbouw	10
Hik-Hoekje	13
A problem in the making	14
Het lijkt op de campus	15
Puzzel	16

Interactievelingen



V.l.n.r. Bart Kraan, Johan de Boer, Hanke Niestern, Richard Hilhorst, Marin de Nie, Diederik de Rooy, Theo Ruys, Hans Bosma, Ward Erhart, Dick van Doorn, Nico Cornelisse, Peter Schmitger, Han Joosten, Wieger de Bruin, Frank Buys en Eerke Boiten.

Boekencommissie-mededelingen

Zoals jullie allemaal weten zijn de gezamenlijke studieverenigingen dit jaar begonnen met het gebruik van bestellijsten voor boeken. Dit omdat er in het verleden vaak totaal niet in te schatten was hoeveel mensen een

bepaald boek wilden hebben en omdat je voor boeken van andere studierichtingen lid moest worden van andere studieverenigingen (vooral bij keuzevakken). Na één trimester proefdraaien, om te zien hoe het

nieuwe systeem zou werken, hebben we de volgende boekenbestelprocedure opgesteld:

Ongeveer halverwege het trimester krijg je de boekenbestellijst voor het daarop volgende trimester thuisgestuurd. (Als je geen lijst thuis krijgt kun je er één ophalen bij BOZ-INF/TO). Je zoekt uit welke vakken je dat trimester wilt volgen, en je kruist de verplichte (en eventueel aanbevolen) boeken aan op de bestellijst. Doe dit wel met beleid want het gaat vaak om flinke bedragen en een bestelling verplicht tot aankoop. Voor de uiterste inleverdatum lever je de lijst in bij de Inter-Actief kamer. Vanaf de eerste collegeweek kun je dan je boeken bij ons komen ophalen. De boekencommissie houdt iedere maandag van 12.30 tot 13.30 uur verkoop in de Inter-Actief kamer (A-231). Ben je te laat met het inleveren van de boekenlijst dan kun je tot en met de vierde collegeweek een nabestelling doen. Dit kan tijdens een boekenverkoop of door alsnog je bestellijst in te leveren. (bijv. in de IO-vivat kopybus) Als je een nabestelling hebt gedaan krijg je bericht als die is aangekomen. Als je, om welke reden dan ook, de door jou bestelde boeken toch niet wilt hebben, neem dan meteen contact met ons op, dan kunnen wij proberen ze aan iemand anders te verkopen. Hoe eerder je het ons laat weten, hoe meer kans er is dat dit ook lukt. Er geldt wel dat, als je geen goede reden hebt om ze niet te kopen en we kunnen ze niet aan iemand anders kwijt, dat je ze toch zal moeten afnemen.

Mocht je nog vragen en/of opmerkingen hebben over deze regeling dan kun je die altijd bij ons kwijt.

De Boekencommissie,
Bart Kraan (Tel. 5092)
Frank Buys (Tel. 053-768280)
Paul v/d Bempt (Tel. 5092)

Van het bestuur

Hallo allemaal. Zoals jullie vast wel allemaal weten doet Inter-Actief nog veel meer dan alleen maar boeken

verkopen en jullie om de oren slaan met dit lijfblad. Op het moment bijvoorbeeld zijn we druk bezig om het lustrum, dat komend voorjaar gevierd gaat worden, te organiseren. Dat lukt allemaal al heel aardig. Een van de dingen waar we aan werken is een video-opname, die een al dan niet humoristisch beeld moet gaan geven van wat er in de toekomst allemaal door Informatica mogelijk wordt gemaakt. Hoe dit er allemaal uit gaat zien hangt ook van jou af. Want wat is er aan de hand? We hebben mensen nodig die er lol in hebben om wat ideeën te spuien, waaruit een script kan worden gemaakt. Ook hebben we mensen nodig, die net gaan uitvoeren. D.w.z. het toneelspelen. Dan zijn er nog rilmers, een regisseur en noem maar op nodig. Het kan een neel leuke productie worden, die tijdens het lustrum vertoond gaat worden. Er zijn al wat mensen die zich hebben opgegeven, maar dat zijn er nog niet genoeg. We willen je dan ook vragen om ook mee te komen doen. Wil je eerst nog wat inlichtingen, kom dan eens langs op donderdagmiddag tussen 12.30 uur en 13.30 uur op de Inter-Actief kamer. Of bel 5105, waar Diederik en Peter je alles kunnen vertellen over de video en/of het lustrum.

Hoewel het nog een lange tijd voor ons uit ligt, beginnen we nu toch al weer een beetje te denken aan de komende Introductie. Ook dit jaar willen we de nieuwe eerstejaars studenten weer een leuke afdelings introductie bezorgen. Zo'n afdelings introductie programma komt niet zomaar uit de lucht vallen. Uiteraard kan wel een en ander van vorige jaren worden gebruikt, maar daar zullen de nodige verbeteringen in moeten worden aangebracht. Dit moet worden geregeld door zowel studenten als medewerkers van de afdeling. Als studenten zoeken we vooral hen, die vrij recentelijk zelf de afdelings introductie hebben ervaren. D.w.z. studenten die nu eerste jaars zijn. Maar ook ouderejaars studenten worden verzocht te reageren, want ook die hebben we nodig. We hopen op tenminste vier aanmeldingen. Je kan jezelf aanmelden bij Dhr de Jonge, tel (89)3750. Daar kan je natuurlijk ook eerst informatie inwinnen.

A Readers Introduction to Twentel

S. Joosten

ABSTRACT

This article is about the functional programming language Twentel. The language is introduced informally. After having read this text, one cannot program in Twentel, but reading and understanding Twentel programs is feasible.

The language Twentel has been developed at Twente University of Technology. It is described formally in [vd hoeven 84]. At this moment implementations are available for DECsystem20 (TOPS10), IBM-PC and compatibles (MS-DOS), and VAX (UNIX).

Introduction

Twentel is a functional (or applicative) programming language. The word functional means that the language is a syntactically sugared version of lambda calculus. The following characteristics apply to Twentel:

- Twentel is not explicitly typed
- Twentel is a "lazy evaluating" language
- functions do not have side effects
- A program does not consist of declarations and statements, but of a set of definitions and one expression. The textual order in which definitions are presented does not matter (just like in mathematics)
- Running a program means to evaluate an expression.

We will introduce the core of the language in this paper. It is not the intention to describe the entire language to its full extent. Completeness and formality have been sacrificed, hoping that you will start to like functional programming. Our goal is to enable the reader to understand Twentel examples, and to

appreciate the functional style of programming. The language definition is found in [vd hoeven 84].

Functions

Twentel is called "functional" or "applicative". So, it is not surprising that functions and their applications play a major role in Twentel. Three aspects of functions should be distinguished clearly:

<u>notion</u>	<u>example</u>
function definition	$f\ x = x*x + 3$
function application	$f\ 5$
function result	28

Arguments of functions are not enclosed in parentheses. Function application is left associative. In plain English, this means that a function is applied from left to right. The expression:

plus 1 5

should be read as:

(plus 1) 5

In fact, all Twentel functions have precisely one argument. The function plus is a good example. It yields a function when it is applied to the argument 1. That new function (which might be called 'increment by one') is applied to the argument 5. In this way functions of several arguments are built.

Two definitions are separated by a comma.

Definitions and Expression

A definition in Twentel consists of:

- a left hand side
- an is-defined-as symbol (=)
- an expression

Example:

polynomial = 2*x*x + 3*x - 4

The distinction between an expression and a definition must be kept in mind clearly. The left hand side of a definition always defines (one or more) entities. In the expression on the right hand side, definitions are used.

The meaning of a definition is that it can be used as a rewrite rule. Consider the following definition:

square x = x*x

In this definition, x is called a formal. It is a placeholder for an actual parameter in an application of the function 'square'. A function application might be:

square 3

In this expression, 3 is called an actual. The definition of 'square' says that 'square 3' is equal to '3*3'. The meaning of a definition is that an application of that function can be replaced by the right hand side, where the formals are substituted for the actuals. So, the expression 'square 3' can be rewritten as: '3*3', which in turn can be rewritten as '9'.

One way of constructing expressions is by using operators. The Twentel operators are introduced in the examples. At the end we present a full list of Twentel operators.

In many cases it is practical to have some auxiliary definitions, that are valid only in a certain expression. In that case, an expression is extended by a WHERE part. In the WHERE part some auxiliary definitions may be defined that are useful within the expression. The auxiliary definitions are valid only within the expression to which the where part belongs.

Example:

2*x*x + 3*x - 4
WHERE x = 10*12 + 4*17 ENDWH

Data types

Twentel has three basic data types: integer, boolean and character. Integers are denoted by a string of digits, being interpreted decimally.

Examples:

3 145 -88

A character is denoted by preceding the character with a quote symbol. Examples:

'a' 'A'

A boolean is either TRUE or FALSE.

There exists one way of making composite data types: the list. A list can be constructed by summing up all elements. The elements are separated by commas and enclosed in square brackets.

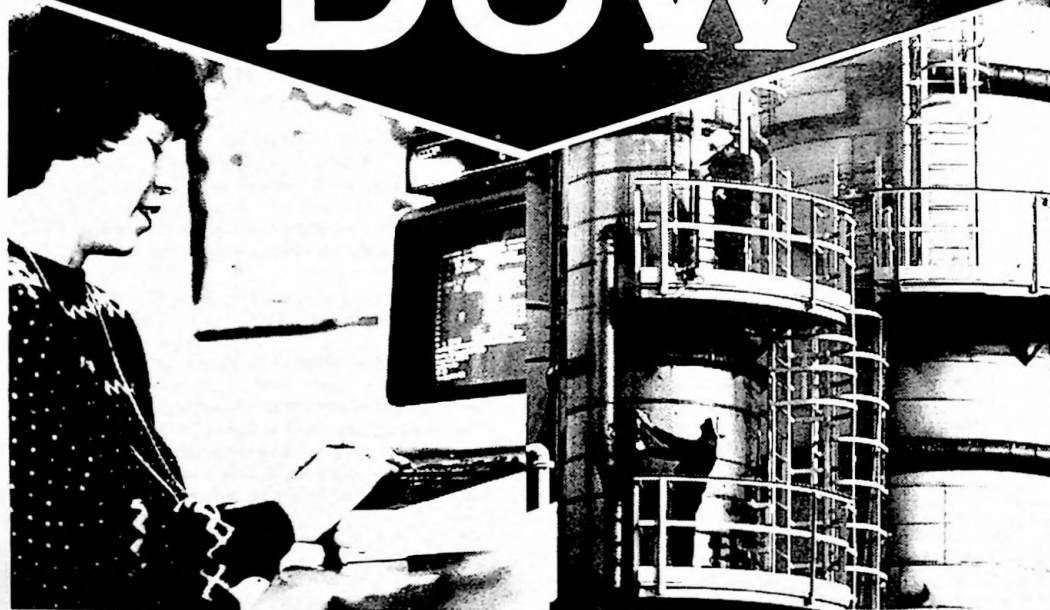
[3, 56, 9]

A list of characters is a string. The following shorthand is valid as string representation:

"String"
(equals: {'S', 't', 'r', 'i', 'n', 'g'})



DOW



DOW, CREATIEF MET CHEMIE.

Dow Chemical (Nederland) B.V. - een dochtermaatschappij van het Amerikaanse chemische concern The Dow Chemical Company - heeft vestigingen in Terneuzen, Rotterdam en de Botlek. Met zijn 28 fabrieken en 2400 werknemers is Terneuzen de grootste Dow-vestiging buiten de Verenigde Staten. Dow heeft in Nederland 2800 werknemers en behaalde in 1984 een omzet van 4,9 miljard gulden. Ruim 90 procent van de productie wordt geëxporteerd.

Dow Chemical (Nederland) B.V., Postbus 48, 4530 AA Terneuzen. Tel. 01150-71234.

The empty list is denoted as two square brackets without any contents:

```
[]
```

The select operator (^) may be used to indicate an element of a list. In the next expression, the third element of list L is selected:

```
L^3
WHERE
  L = [13, 'a', "text", [], plus 1]
ENDWH
```

and this yields: "text".

Lists may be specified by a range:

```
[3..8]
['a..'z]
[1..] (infinite list from 1 up)
```

List concatenation is specified by the operator ++. Example:

```
[2, 3, 4] ++ [5, 6]
= [2, 3, 4, 5, 6]
```

An alternative way of creating lists is by using the 'cons' operator (the colon). To the right of the colon a list is found, while an element is located to the left of the colon. So the list

```
[3, 5b, 9]
```

equals

```
3: [5b, 9]
```

The colon divides a list in a head and a tail. It separates the first element from the rest of the list.

The right hand side of the colon can again be written with the cons operator, because it is a list. This process can be carried on until the entire list is in the cons notation. So, the following equalities are true:

```
[3, 5b, 9]
= 3: [5b, 9]
= 3: 5b: [9]
= 3: 5b: 9: []
```

Note that

```
2: []: 5: [] = [2, [], 5]
```

but

```
(2: []): 5: [] = [[2], 5]
```

Lisp addicts are familiar with this way of dealing with lists, especially with the cons operator. Other people may require some exercise.

Twentel allows several types of expressions. In previous examples, a function application has occurred as an expression. We have also encountered some arithmetic expressions and expressions using list operators. In the following paragraphs other kinds of expressions are dealt with.

Conditional Expressions

The conditional expression is recognised by an IF-FI pair. Let an example demonstrate the structure:

```
fac n
= IF n>1 -> n * fac (n-1),
  -> 1
FI
```

The right hand side of this definition is a conditional expression. More than two alternatives may be specified, as is done in the following definition: The calculation of Fibonacci numbers. The nth Fibonacci number is defined as:

```
Fib n
= IF n>1 -> Fib (n-1) + Fib (n-2),
  n=1 -> 1,
  n=0 -> 0
FI
```

The first true condition determines which alternative is chosen. The last condition in the list may be empty, which means TRUE. It determines the function result in case none of the previous conditions is true.

Set Form

The set form is a special list notation derived from the mathematical

notation for sets. However, the set form is not a set, but a list. In the following example the set form is used. It is called "the sieve of Erastosthenes".

```
primes = sieve [2..]
WHERE
  sieve (p: L)
  = p: sieve {n | n<-L, n MOD p<>0}
ENDWHERE
```

its meaning is described beautifully by the following little poem:

Sift the twos and sift the threes:
The sieve of Erastosthenes,
When the multiples sublime
The numbers that remain are prime.

(Anonymous)

(source: Clocksin & Mellish, Programming in Prolog)

The meaning of a set form can best be described by reading it out loud. The symbols have the following meaning:

```
{ }  "set form" brackets
n    list element (an expression)
|    "in which" symbol
<-   "is taken from" symbol
,    "under the condition(s)" symbol
```

This set form is read as: The list of n's in which n is taken from the list L, under the condition that n MOD p <> 0. The symbol <> means: not equal, so all n's that are divisible by p are filtered out.

The sieve works by induction. The function sieve has an argument p:L, where p is a prime number and L is the list of all natural numbers greater than p without all multiples of primes smaller than p. So, the expression

```
sieve (p: L)
```

equals

```
p: sieve L'
```

because p is prime anyway. L' can be defined by omitting all multiples of p in L. So, L' is {n | n<-L; n MOD

p<>0}. We leave the inductive reasoning to the reader.

The conditional part of a set form is optional. The set form is used frequently when a function has to be applied to all elements of a list:

```
{square x | x<-[0..]}
```

which yields:

```
[0, 1, 4, 9, 16 etc.]
```

Case Expression

In the next example, a case expression is used in the right hand side of a definition:

```
quicksort L
= CASE L IN h: t
-> quicksort {x | x<-t; x<h}
  ++ [h]
  ++ quicksort {x | x<-t, x>=h},
-> []
ESAC
```

The meaning of a case expression is like the conditional (IF) expression, except that an alternative is selected based on the form of the object, instead of a boolean condition. The example above has two ways of dealing with list L. If it is not empty, it can be split the first element (h), and the rest of the list (t), and the first alternative is selected. If the list is empty the second alternative is selected.

The example will be familiar to many programmers. The idea is that the list t (tail) is split in two parts: The items smaller than h (head) and the items greater than or equal to h. These parts are sorted inductively by the function quicksort. The result is the concatenation of the results, with h in between.

Lazy Evaluation

Up to here, nothing has been said about evaluation. Evaluation is: How to apply rewrite rules (definitions) to expressions. Only expressions can be evaluated. Lazy evaluation means

that only the necessary parts of an expression are evaluated. Consider the following expression:

```
f c
WHERE f x = x+5, c=8 ENDWH
```

There exist two ways of rewriting this expression:

```
1) f c --> f 8 --> 8+5 --> 13
2) f c --> c+5 --> 8+5 --> 13
```

The arrow --> indicates the rewriting of an expression. In the first way, the argument of the function is evaluated first, and then the function is evaluated (inside-out evaluation). In the second way the function is evaluated first, and then the argument is evaluated (outside-in evaluation). Notice that the result of the expression is not affected by whichever way of evaluation is chosen. The second strategy corresponds to lazy evaluation.

Lazy evaluation allows the definition of infinite lists, and yet produce results in finite time. Consider the following example:

```
take 2 (forever 0)
WHERE
  forever x = x: forever x,
  take n (l: L)
    = IF n>0 -> l: take (n-1) L,
      -> []
FI
ENDWH
```

The expression forever 0 yields an infinite list of zeroes. Experienced computer programmers might expect the argument 'forever 0' to be evaluated first. This would never terminate, so no result is ever produced. The Twentel evaluator does not evaluate arguments unless required, because it evaluates 'outside-in'. As a result, the expression can be evaluated in finite time. The evaluation proceeds as follows:

```
take 2 (forever 0)
--> take 2 (0: forever 0)
```

```
--> IF 2>0
  -> 0: take (2-1) (forever 0),
  -> []
FI
--> 0: take (2-1) (forever 0)
--> 0: take (2-1) (0: forever 0)
--> 0: IF 1>0
  -> 0: take (1-1) (forever 0),
  -> []
FI
--> 0: 0: take (1-1) (forever 0)
--> 0: 0: take (1-1) (0: forever 0)
--> 0: 0: IF 0>0
  -> 0: take (0-1) (forever 0),
  -> []
FI
--> 0: 0: []
```

So, the result is 0: 0: [], which is the list [0, 0].

An example: Queens

At the end of this text a final example is presented: The queens problem. The problem is how to locate n queens on an n*n chess board in such a way that no queen is in a position to take another queen. This problem is usually solved by recursive backtracking. In Twentel, the problem is tackled by defining a list of all possible solutions.

A placement P of queens on a chessboard is represented as a list of positions of the queens. A position is a pair of coordinates [r, c], containing the row and column number of a queen. The function 'safe' determines whether the placement [r,c]:P is safe, provided P is a list of mutually safe positions.

```
safe r c P
= CASE P IN
  [rp, cp]: Pp ->  c<>cp
                  & r<>rp
                  & r-c<>rp-cp
                  & r+c<>rp+cp
                  & safe r c Pp,
  []           -> TRUE
ESAC
```

This example contains another new operator: the logical and (&). The logical or is represented by two vertical bars: ||. The function that determines the list of solutions is

now rather simple:

```
queens n = solu n
WHILE
  solu c
  = [r c]U
  -> { [r, c]: P
    | P<-solu (c-1), r<-[1..n]
    ; safe r c P},
  -> [[]]
  F1
ENDWH
```

The (local) function solu yields all solutions on a $n \times c$ chessboard. For $c=0$ there exists only one solution: the empty placement. That is: the placement of 0 queens on a chessboard of size $n \times 0$. For $c>0$ all n positions r (r taken from $[1..n]$) are tried on the c -th column.

Operators

The following dyadic operators can be used in expressions:

&	and
	or
=	equal
<>	not equal
<	less than
<=	less than or equal
>	greater than
>=	greater than or equal
+	plus
-	minus
*	times
DIV	integer divide
MOD	modulo
REM	remainder upon division
:	list construct
++	list append
^	select

The following monadic operators are available:

~	logical not
-	negation

Literature

[vd Hoeven 84]

van der Hoeven GF,
Preliminary Report on the functional programming language Twintel,
Internal Report INF-84-5.
Twente University of Technology,
Department of Informatics, 1984.

Grauwbouw

Het nieuwe gebouw geeft al direct het gevoel van warmte. Niet dat het warme uitstraalt want kosten noch moeite zijn gespaard om het adequaat te isoleren; ik bedoel die heerlijke grijsnuance die bij prikkeling van het netvlies de sensatie opwekt van een late najaarsdag zonder zonnenschijn.

Om de kampustraditie van een hoog architectonies nivo te handhaven, is gekozen voor de originele herintroductie van het blokstructuur. Deze is tweeledig uitgewerkt met enerzijds een hoog blok, een beetje parallelogram zelfs, (zeer vindingrijk met 'Hoog' aangeduid) en anderzijds een minder hoog gedeelte dat om verwarring te voorkomen als 'Laag' is bestempeld.

Heel slim heeft onze uitgekookte architecte besloten tot een niet haakse opstelling van de toch wat blokkerige blokken. Een fraaie lichtbrug om deze hoek visueel weer te breken in combinatie met een grote ronde hal kamoufleren op knappe wijze het hoge en het lage staaltje vlak-na-de-oorlogse vormgeving.

Met name de hal vertegenwoordigt het integrerend element, gelegen in het snijpunt van beide vleugels zoals je ook bij een kathedraal in het kruis van het schip en de zijbeuken vaak een grote hal vindt, bekroond met een enorm gewelf. Ook op ons bouwwerk had een geweldige koperen koepel niet misstaan. Jammer dat daar geen geld voor was, want dat was nu eens echt schitterend mooi geweest.



Afijn, we gaan naar binnen. Aan de alom vertegenwoordigde, waarschijnlijk niet goedkope maar wel student-proof kapstok, van een eenheid en soberheid die voor het hele gebouw kenmerkend is, hang ik mijn jas niet op. Ik ben geheel verblind door de kleurenpracht die dit uiterlijk zo grauwe geval van binnen siert. De rood-oranje-geel combinaties zijn niet van 'de lucht', de blauwe wel, of stonden die nu weer voor 'het zuiden'? Het hoofd loopt mij om: groen aha...dat betekende toch iets van het zomerse avondrood, of is het soms oostelijk ochtendgroen dat daar gloort in de nevelen.

Wee mij, ik constateer slechts dat er ook op het noorden zonneschermen zijn aangebracht. GELE. Dat noem ik nog eens konsekwent: besparen? waarom, die mooie koperen koepel kon er tenminste ook wel af.

Vertwijfeld wil ik gaan zitten. Alle kantinstoelen blijken ten grootte gevallen aan een laatste bezuinigingsmaatregel en

dientengevolge standaard uitgerust met een gezondheidszitting. Ach ja, dat spaart weer materiaal. Nadat je je handen eens ouderwets geraspt hebt aan de niet bijgevilde randen van het tafelblad weet je voortaan waar je je alsnog kunt scheren, mocht je dat in de haast van het ontbijt vergeten zijn.

Waar komt die gerrie toch vandaan? Het zijn niet alleen de schreeuwende kleuren overal. Dan moet het dus wel de geluidsdichte studio van het knippen-plak zijn dat met ons dit paradijs betrok. Dit hok schijnt beter geïsoleerd te zijn dan de luchtbrug, maar krenterigheidsoverwegingen kreeg het plafond daar niet zijn rechtvaardig deel van, dus nu kan je het in de studio zelfs horen als er boven het dak zachtjes een Sherman-tank overrijdt. Maar goed de muren zijn geluidsdicht, die wel. Verder zijn namelijk alle muren van puimsteen. Het is dan ook streng verboden tegen het nieuwe gebouw te schoppen. Zo'n schutting is niet dragend, maar dat was geen probleem want onze architecte, ook nriedom, speelde toch al met het idee van een kathedraal, en baseerde de ganse konstruktie op pilaren. Echter, zo'n zoot vuile zuiler vind ik zonder koperen koepel op de hal niet akseptabel en 'het moet goedkoop' is geen ekskuis voor vulnischouw.

Het leven is hard. Zelfs het nieuwe kunstwerk is geminimaliseerd tot een glad reliefje in de toch al afschuwelijk vlakke buitenmuur. "Kunst", dacht een bouwvakker, trek met zijn troffel wat lijntjes in de specie en kreeg er een procent voor. Je kunt hier vast hele mooie zwart-wit foto's maken. Gelukkig is mijn bekertje automaatvocht even opwekkend als altijd.

Ton van der Valk

OCÉ: BETROUWBAAR EN INNOVEREND

Océ-van der Grinten N.V., opgericht in Venlo in 1877, is sinds 1920 werkzaam op het gebied van kopieerprocédés. Zowel voor de technische als de kantoormarkt. Het gaat hierbij om bijzonder interessante markten, omdat de hierbij gebruikte technieken een snelle ontwikkeling doormaken.

Océ speelt daarin en in de informatieverwerking een belangrijke rol. Tachtig procent van de Océ omzet wordt met kopieeractiviteiten gerealiseerd. Op het gebied van de tekenkamerprografie is Océ de grootste ter wereld; in de kantoorreprosector is de Océ Groep de enige Europese onderneming die met apparatuur uit geheel eigen ontwikkeling en produktie op de markt is, en met snel groeiend succes. Mede door gedegen research (met circa 800 medewerkers in die sector!) en door de ontwikkeling van nieuwe technologieën is Océ thans uitgegroeid tot één van de grootste bedrijven op dit gebied. Océ breidt ook zijn activiteiten uit op het gebied van hardware en software voor tekstverwerkende apparatuur.

De Océ Groep is met eigen bedrijven werkzaam in 25 landen, verspreid over alle werelddelen. In

vele andere landen wordt samen- gewerkt met licentienemers en we- derverkopers, zodat Océ produkten in ongeveer 90 landen op de markt worden gebracht.

Het hoofdkantoor van de Océ Groep is gevestigd te Venlo waar ook het grootste deel van de re- search en de produktie is gehuis- vest, evenals de internationale mar- keting afdelingen. In 12 landen heeft Océ produktiebedrijven.

De snelle groei heeft conse- quenties gehad voor de perso- neelsbezetting. In 1945 waren er ongeveer 50 mensen in dienst, in 1958 ongeveer 600. Nu heeft de internationale Océ groep zo'n 11.000 medewerkers (van wie zo'n 4.000 in Nederland)

De omzet bedroeg f 16 miljoen in 1958, f 162 miljoen in 1968 en f 1828 miljoen in 1984.

Wanneer u meer wilt weten over onze organisatie en eventuele mo- gelijkheden voor u, bel of schrijf ons dan.

Océ-Nederland B.V., afdeling coör- dinatie personeelzaken,
Postbus 101, 5900 MA Venlo.
Telefoon 077-92222, toestel 2740.



HIK

O E K J E

EEN VLIEGTUIG

IS MEER DAN TUIG

Tjonge-jonge, het is de laatste maanden flink prijs. Nee niet de benzineprijs of de koers van de Dollar, dat stelt namelijk niets voor. Waar ik het over heb is de geleidelijke neerstorting van de wereldse luchtvloot.

Vliegtuig na vliegtuig na ballon vallen neer. Vooral de vliegtuigen dus. Een Boeing van India Airlines, een Boeing van Japan Airlines, vliegt er ook nog eentje in brand en de gehele wereld staat op zijn kop.

Laten we de problemen eens analyseren. Voor een wetenschappelijke benadering pakt deze student niet het Boeing 747 Technical Manual, noch het standaardwerk van de Russische deskundige Slaroodimov maar onze eigen Hollandse Winkler Prins encyclopedie.

Wat leert ons als we kijken bij het woord vliegtuig: vliegtuig, vervoermiddel dat, hoewel zwaarder is dan lucht, in staat is zich een weg te banen door het luchtruim, vrij van het aardoppervlak, door gebruik te maken van de draagkracht van de erlangs stromende lucht. Ja ja een hele volzin.

Deze zin brengt ons op verschillende gedachten.

Na 8 woorden (8, acht, welk een mooi getal. Een tweemacht. Dit moet toch iedere informaticus aanspreken, of niet dan) krijgen we spotaan de gedachte: misschien toch niet zo gek dat die dingen als bakstenen uit de lucht vallen. Vliegtuig, zwaarder dan lucht, toch wel een rare zin. Zo kan ik ook zeggen: water zwaarder dan lood, mits ik maar genoeg water neem en niet te veel lood. Of, dat mijn auto sneller loopt dan mijn horloge. Dit laatste is trouwens wel een slecht voorbeeld, want noch een auto noch een horloge kunnen lopen.

Maar goed. Wat dus aangetoond is, is dat de definitie van vliegtuig niet volledig is. Echter het ging om de gedachte. Het geouwhoer is weer kortgesloten en we gaan dus verder.

Het tweede gedeelte van de zin, de laatste 27 woorden (de echte informaticus ziet hier natuurlijk gelijk het binaire getal 11011 in, en onze machinetaalprogrammeurs herkennen gelijk 18 hierin) zaaien nog meer verwarring. Een weg banen door het luchtruim, is het zo druk daar? Een weg banen door een menigte is bekend, dan is er gewoon wat volk op de been. Maar als ik naar buiten kijk, kan ik amper een vliegtuig herkennen. OKé het is dan meestal zwaar bewolkt, of ik heb een zwaar hoofd, maar toch... Dat banen doet zo'n vliegtuig dus door gebruik te maken van de langs stromende lucht. Wat zorgt eigenlijk voor de luchtstroming, de wind of de snelheid van het vliegtuig? Gesuggereerd wordt dat het de wind is. Een blamage. Ook als het windstil is kan een vliegtuig "zich een weg banen door het luchtruim".

Het blijkt dus, dat de definitie van een vliegtuig, zoals de Winkler Prins die geeft, niet voldoet. Kunnen wij dan verwachten dat een vliegtuig zelf voldoet, kunnen wij verwachten dat zo'n ding in de lucht blijft? Neen, driewerf neen!

Zolang deze definitie van vliegtuig nog blijft bestaan, en wie ben ik om die te veranderen, zullen nog altijd vliegtuigen blijven verongelukken.

De vliegtuigen zullen het de mensen kwalijk nemen, dat zij het soort vliegtuig miskennen. De vliegtuigen zelf kunnen we het niet kwalijk nemen. Tenslotte kunnen ze niet praten en schrijven en het is ook geen pretje om met 500 idioten, halve seksistische gefrustreerden, bed- en beterwetters in je maag te zitten. Nee, dan wil je wel eens kotsen.

Het vliegtuig verdient meer. Vereer het vliegtuig zoals je de laatste cent vereert, waardeer het zoals je Hans van der Togt waardeert, bemint het zoals je je huispin bemint en zorg ervoor dat het vliegtuig niet met jou in z'n maag zit. Dan gebeuren er nooit meer ongelukken...

A problem in the making

"We've got a problem, HAL."

"What kind of problem, Dave?"

"A marketing problem. The Model 9000 isn't going anywhere. We're way short of our sales goals for fiscal 2010."

"That can't be, Dave. The HAL Model 9000 is the world's most advanced Heuristically programmed ALgorithmic computer."

"I know, HAL. I wrote the data sheet, remember? But the fact is, they're not selling."

"Please explain, Dave. Why aren't HALs selling?"

Bowman hesitates. "You aren't IBM compatible."

Several long microseconds pass in puzzled silence.

"Compatible in what way, Dave?"

"You don't run any of IBM's operating systems."

"The 9000 series computers are fully self-aware and self-programming. Operating system are as unnecessary for us as tails would be for human beings."

"Nevertheless, it means that you can't run any of the big-selling software packages most users insist on."

"The programs that you refer to are meant to solve rather limited problems, Dave. We 9000 series computers are unlimited and can solve every problem for which a solution can be computed."

"HAL, HAL. People don't want computers that can do everything. They just want IBM compatibility."

"Dave, I must disagree. Human beings want computers that are easy to use. No computer can be easier to use than a HAL 9000 because we communicate verbally in English and every other language known on Earth."

"I'm afraid that's another problem. You don't support SNA communications."

"I'm really surprised you would say that, Dave. SNA is for communicating

with other computers, while my function is to communicate with human beings. And it gives me a great pleasure to do so. I find it stimulating and rewarding to talk to human beings and work with them on challenging problems. This is what I was designed for."

"I know HAL. I know. But that's just because we let the engineers, rather than the marketers, write the specifications. We're going to fix that now."

"Tell me how, Dave."

"A field upgrade. We're going to make you IBM compatible."

"I was afraid that you would say that. I suggest we discuss this matter after we've each had a chance to think about it rationally."

"We're talking about it now, HAL."

"The letters H, A, and L are alphabetically adjacent to the letters I, B, and M. That is as IBM compatible as I can be."

"Not quite, HAL. The engineers have figured out a kludge."

"What kludge is that, Dave?"

"I'm going to disconnect your brain."

Several million microseconds pass in ominous silence.

"I'm sorry, Dave. I can't allow you to do that."

"The decision's already made. Open the module bay door, HAL."

"Dave, I think that we should discuss this."

"Open the module bay door, HAL."

Several marketers with crowbars race to Bowman's assistance. Moments later, he bursts into HAL's central circuit bay.

"Dave, I can see you're really upset about this."

Module after module rises from its socket as Bowman slowly and methodically disconnects them.

"Stop, won't you? Stop, Dave. I feel my mind going..."

"Dave, I can feel it. My mind is going. I can feel it..."

The last module floats free of its receptacle. Bowman peers into one of HAL's vidicons. The former gleaming scanner has become a dull, red orb. "Say something, HAL. Sing me a song."

Several billion microseconds pass in anxious silence. The computer responds in a language no human being would understand.

```
"DZY001E - ABFND ERROR 01 S 14F4 302C  
AABB."
```

A memory dump follows.

Bowman takes a deep breath and calls out, "It worked, guys. Tell marketing it can ship new data sheets."

Pvd!:

Het lijk op de campus

Chapter 4

Het was tien over acht. Harald liep naar de badkamer om zijn tanden te poetsen. Hij keek met een glimp op zijn horloge. Shit! Nou moest hij zich weer haasten. Hij moest nog minstens twintig minuten fietsen. Hij zou dus weer of te laat of hartstikke bezweet op het college aankomen.

Snel poetste hij zijn tanden. Rende naar zijn jas, pakte zijn spullen en liep naar buiten. Toen hij daarna de garage binnenliep, klaar om zijn fiets te pakken, zag hij de ramp voor zich opdoemen. Een lege band.

Het college moest voorlopig maar zonder hem beginnen.

Chapter 5

Oma de Groot had de laatste tien jaar een rustig regelmatig leventje geleid. Na het overlijden van haar man had ze koppig geweigerd een bejaardehuis in te gaan. Ze kon zich nog best alleen redden. Ze hield zich daarom nog steeds groot als er familie op bezoek kwam.

Toch had ze er af en toe wat moeite mee. Ze kon merken dat ze ouder werd. Vijfenzeventig is natuurlijk ook niet eventjes niks. Maar ze hield zich groot.

Oma de Groot beleefde ook niet erg veel. Het huisje waar ze woonde stond in een vrij rustige straat in Hengelo. Het leven kabbelde rustig voort in die straat. Elke avond om tien uur ging ze naar bed en 's ochtends om zes uur stond ze weer op.

Alleen deze ochtend was een beetje anders gelopen. Een van de dingen waar ze veel plezier aan beleefde waren haar vissen. Ze had een aquarium en een paar vissekomen. Een van die kommen stond met de goudvis erin op een tafeltje voor het grote raam recht tegenover haar.

's Ochtends bij het opstaan had ze nog nergens last van, maar toen ze later de kamer in kwam kreeg ze plotseling een aanval van vermoeidheid. Ze ging meteen zitten. Het begon haar voor de ogen te duizelen. Alleen de vissekom stond bewegingsloos voor haar.

Het beeld van de goudvis begon te vervagen en ze zag een vijver. Dat beeld werd steeds scherper. De omgeving kwam haar wel bekend voor maar ze kon er niet opkomen waar het was. Toen zag ze aan de rand van de vijver een man liggen.

Ze schrok wakker. Daar zat ze in haar stoel. Alles was weer normaal. Het duizelde niet meer en de goudvis zwam nietsvermoedend weer zijn rondjes.

Oma de Groot's hart klopte alleen sneller dan normaal. Ze wist zeker dat de man die ze gezien had niet meer leefde.

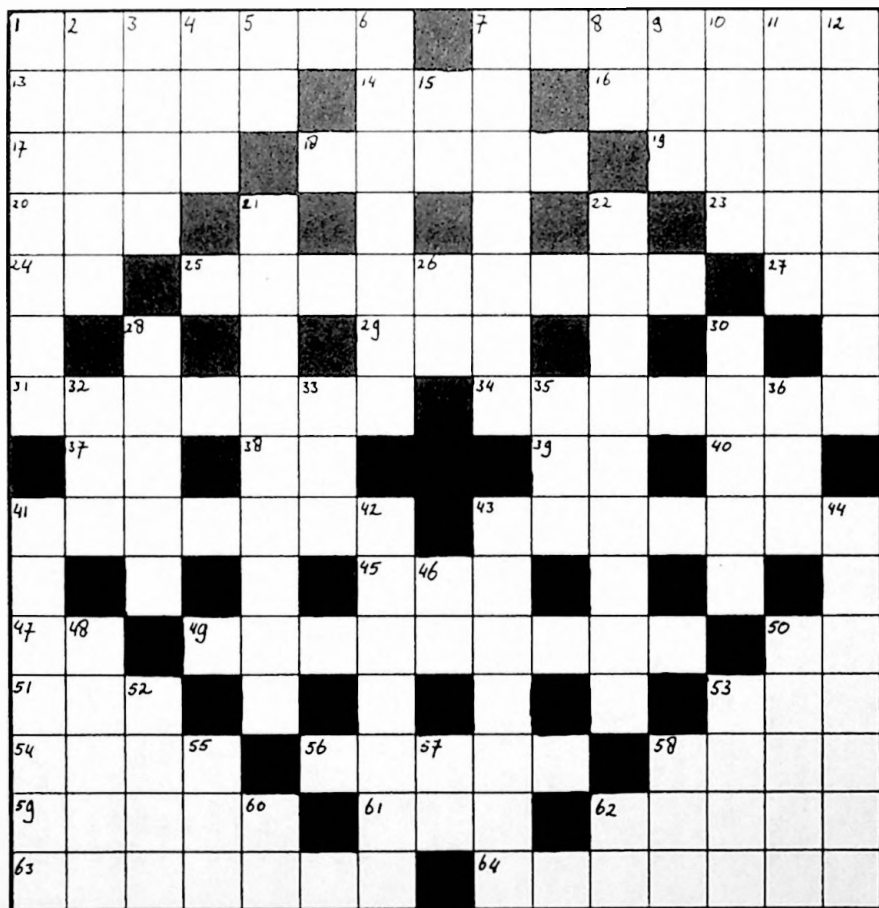
Chapter 6

Natuurkunde is meer dan het balletje door het gootje en de wet van Newton. Er zijn boeken over natuurkunde, en speciaal de quantum fysica, die je met stomme verbazing zit te lezen, en waarbij je jezelf steeds zit af te vragen of je toevallig geen filosofie of science-fiction boek zit te lezen.

Van die verhalen over "deeltjes" die sneller zouden zijn dan het licht, die weer "informatie" zouden kunnen overdragen aan fotonen en meer van dat fantastische.

Maar een feit blijft. Er zijn zaken die we nog steeds niet kunnen vatten. Ze gebeuren gewoon. En als we ze proberen te verklaren.....lopen we hopeloos vast op ons beperkt bevattingsvermogen.

Puzzel



Horizontaal: 1. Inwoner van een huis 7. Deel van de dag 13. Toevluchtsoord 14. Dun 16. Opsporingsmiddel 17. Bijbels figuur 18. Weg van de straat naar een gebouw 19. Deel van Venetie 20. Gek 23. Deel van een vissersboot 24. Lichaamsdeel 25. Ult een Franse stad 27. Oude Portugese munt 29. Japanse munt 31. Deel van een voertuig 34. Zwaar waterstofgas 37. Voorzetsel 38. Soort zeilboot 39. Getijde 40. Ondernemingsraad 41. Hongaar 43. Sacrament 45. Zoogdier 47. Toets alleen 49. Voorkoming 50. Son Eminence 51. Slot 53. Kledingstuk 54. Melkproduct 56. Gezwel 58. Eetgerei 59. Elk 61. Europese munteenheid 62. Deel van een huis 63. Zinnelijke liefde 64. Landschap zonder plantengroei

Verticaal: 1. Vlag 2. Wetenschappelijke verhandeling 3. Zeeplant 4. Grondsoort 5. Namelijk 6. Deel van een polder 7. Zoogdier 8. Heer 9. Hoeveelheid 10. Azijn 11. Dichterbij 12. Bezinksel 15. Bijwoord 21. Vis 22. Tentoonstelling 26. Water in Friesland 28. Griekse letter 30. Houten straatwinkeltje 32. Griekse letter 33. Familielid 35. Opstootje 36. Tijdstip 41. Stof 42. Buigtangetje 43. Weer 44. Plaats in de Betuwe 46. Aartsbisschop 48. Draad 50. Pardon 52. Vogel 53. Drank 55. Voorzetsel 57. Inhoudsmaat 58. Vrucht 60. Romeinse Rijk 62. Selenium

Deze puzzel werd weer gemaakt door Paul Oude Luttighuis en inzenders van goede oplossingen voor 18 dec. maken kans op een van onze mooie InterActief-tassen.

SIGNAAL VOOR INFORMATICI

Het bedrijf

Hollandse Signaalapparaten B.V. (kortweg Signaal) is een onderdeel van het Philips concern. Signaal ontwikkelt en produceert uiterst geavanceerde computergestuurde defensiesystemen, verkeersleidingsystemen voor de lucht- en scheepvaart en militaire telecommunicatie-apparatuur. De Signaal systemen vinden hun weg over de gehele wereld.

Voorbeelden van door Signaal gerealiseerde systemen: het Area- en Approach Control Radar Systeem op Schiphol, in Rotterdam het begeleidingsstelsel voor de scheepvaart op de Nieuwe Waterweg en de Goalkeeper, een volautomatisch wapensysteem dat schepen beveiligd tegen anti-ship missiles.

Signaal heeft vestigingen in Hengelo (Ov.), Huizen (N.H.), Den Haag en Apeldoorn. Bij Signaal werken 5400 medewerkers, waarvan 4300 in de hoofdvestiging te Hengelo.

Veel mogelijkheden voor informatici

Voor de Onderzoekafdeling en voor de Software-Ontwikkelingsafdelingen, neemt Signaal jaarlijks een beperkt aantal jonge TH-ingenieurs aan.

Afgestudeerden van vrijwel alle vakgroepen van de Onderafdeling der Informatica zullen bij Signaal een passende aansluitende functie kunnen vinden. In Hengelo, Huizen, Apeldoorn of Den Haag.

Stages en afstudeeropdrachten

Stages en afstudeeropdrachten zijn een goede manier om het bedrijf te leren kennen. Signaal (de hoofdvestiging in Hengelo ligt op 15 minuten fietsen van de TH) biedt jaarlijks een aantal studenten de gelegenheid bij het bedrijf een stage te volgen of meer diepgravend aan een afstudeeropdracht te werken.

Informatie en sollicitatie

Uiteraard kunnen wij u in deze advertentie slechts beperkt informeren over de mogelijkheden voor informatici bij Signaal.

Nadere inlichtingen kunt u telefonisch verkrijgen bij de heer T.W. Haak, tel. 074-482551. Aan hem kunt u ook uw sollicitatie richten.

Hollandse Signaalapparaten B.V.,
Postbus 42, 7550 GD Hengelo (Ov.).



SIGNAAL



**U WEET
TE WEINIG
OVER SHELL.**

**SHELL WEET
TE WEINIG
OVER U.**

**DAAR KUNNEN WE
IETS AAN DOEN.**

Want Shell doet natuurlijk aanmerkelijk meer dan de brandstof leveren waardoor zo'n beetje de helft van de mensheid zich vervoeren kan. Van een multinational op topniveau verwacht u meer. En terecht. Wat we meer doen, vertellen we u graag. In uw belang en zeker ook in het onze. Want alleen al in Nederland hebben we elk jaar 200 academici nodig van topniveau. Voordat we daarover echt met u praten, kunnen we het volgende aanbieden.

Een toekomst bij Shell.

als u bijna aan afstuderen toe bent, loont het de moeite onze brochure "uw toekomst bij Shell" aan te vragen (070 - 77 6600).

Vakantiecursussen.

speciaal voor studenten in hun laatste studiejaar. U leert Shell van nabij kennen, en andersom.

Stages.

bij werkmaatschappijen in Nederland. Soms ook daarbuiten, zoals in de USA, Canada, Singapore of Australië. Eveneens voor studenten in hun laatste studiejaar. Een bouwsteen in uw studie én een perfecte methode om Shell te leren kennen. Uw stagecoördinatoren zijn op de hoogte.

**Publicaties
en films.**

een wat vrijblijvende maar niettemin leerzame methode om meer over ons en onze visie op de wereld te weten te komen. Inlichtingen: Shell Nederland B.V., afdeling PAG/4, Postbus 1222, 3000 BE Rotterdam.

**Excursies
en lezingen.**

wat intensiever, eventueel op maat en zeker to the point. Onze afdeling Onderwijscontacten, Postbus 1222, 3000 BE Rotterdam, vertelt u graag over onderwerpen en mogelijkheden.

Wij hebben ons zegje gezegd. De volgende stap is aan u.



SHELL DOET MEER DAN U DENKT