Copyright Notice

The following manuscript,

EWD 1017: Fillers at a YoP-institute

was published as

Edsger W. Dijkstra, ed., *Formal Development of Programs and Proofs*. Addison-Wesley, 1990: 209–227.

Fillers at a YoP-institute

Fermat and Wilson

From graph theory, we use

- a finite, directed graph in which each node has both in-degree and out-degree $=1$ consists of cycles;
- consider along a cycle a path of p edges; if the path ends at the node at which it starts, the length of the cycle – i.e. the number of nodes on it – is a divisor of p; in particular, if p is prime, the cycle is of length p or is of length 1 (i.e. is a "self-loop").

We can use this to prove for natural n and prime p

$$(n^p - n) \bmod p = 0 \qquad \text{(Fermat)}$$

$$((p-1)! - (p-1)) \bmod p = 0 \qquad \text{(Wilson)}$$

For the proof of the theorem of Fermat we take as nodes the n^p strings of p characters from an alphabet of size n, and introduce a directed edge from Rr to rR for any character r and string R (of length $p-1$). According to ••, the graph consists of cycles of length p and of self-loops. Because the self-loops correspond to the strings in which all characters are the same and because the size of the alphabet is n, the number of nodes occurring in self-loops equals n. The remaining $n^p - n$ nodes are therefore partitioned into cycles of length p, which proves Fermat's theorem.

For the proof of the theorem of Wilson we take as nodes the $(p-1)!$ cyclic arrangements of the numbers from 0 through $p-1$; for each directed edge we obtain (the cyclic arrangement corresponding to) the target node by increasing each number in (the cyclic arrangement corresponding to) the source node by 1 modulo p . According to ●● , the graph consists of cycles of length p and of self-loops. Because the self-loops correspond to the cyclic arrangements with constant difference (modulo p) between adjacent numbers, and because 1 through $p-1$ are the possible values of that difference, the number of nodes occurring in self-loops equals $p-1$. The remaining $(p-1)!-(p-1)$ nodes are therefore partitioned into cycles of length p , which proves Wilson's theorem.

Maximizing the product for given sum

Question How do we construct a bag of positive integers with given sum so that their product is as large as possible ?

Answer Because with sum ≤ 1 , the bag is unique, we only analyse the cases with sum ≥ 2 .

- because $1 \cdot x < 1+x$, and (because of sum ≥ 2) our target bag differs from $\{1\}$, our target bag contains no 1 ;
- because $2 \cdot (x-2) \geq x \equiv x \geq 4$, our target bag need not contain integers ≥ 4 ;
- because $2+2+2 = 3+3$ and $2 \cdot 2 \cdot 2 < 3 \cdot 3$, our

target bag contains at most two 2's.

The above constraints are met by a bag of (2·sum) mod 3 2's and for the rest 3's . (End of Answer.)

Remark The predominance of 3's in the target bag reflects that 3 is the closest integer approximation of e – the base of the natural logarithm – , which is the solution of the corresponding continuous problem. For the same reason it is theoretically preferable to implement Heapsort with a ternary tree instead of a binary one. (End of Remark.)

On "Poor Man's Induction"

We consider n points along the circumference of a circular cake and cuts along all the chords between them, the points being chosen in such a way that all internal intersection points of pairs of chords are distinct. With f the number

etc

n=	1	2	3	4
f=	1	2	4	8

of pieces, the above suggests $f = 2^{n-1}$. The reader may verify that n=5 indeed yields f=16. However, n=6 yields f=31 ! How does f depend on n ?

This problem is most easily solved in two steps:

the first step expresses f in terms of

c = the number of chords, and
p = the number of internal intersection points,

the second step expresses c and p in terms of n.

- for the increase Δf of f caused by a new chord we observe

Δf
= {the new chord cuts pieces into two}
the number of pieces cut by the new chord
= {a piece is cut by a segment of the new chord}
the number of segments on the new chord
= {segments are separated by intersection points}
1 + the number of intersection points on the new chord
= {internal intersection points of pairs of chords are distinct}
$\Delta c + \Delta p$.

From this and $c=0 \Rightarrow p=0 \wedge f=1$ we derive by mathematical induction

$$f = 1 + c + p \quad .$$

- from the one-to-one correspondence between chords and pairs of points on the circumference we derive

$$c = \binom{n}{2} \quad ;$$

from the one-to-one correspondence between internal intersection points and quadruples of points on the circumference we derive

$$p = \binom{n}{4} \quad .$$

Combining the results from ●● we establish

$$f = 1 + \binom{n}{2} + \binom{n}{4}$$

or -by properties of binomial coefficients- equivalently

$$f = \binom{n-1}{0} + \binom{n-1}{1} + \binom{n-1}{2} + \binom{n-1}{3} + \binom{n-1}{4} \quad ,$$

i.e. f equals the sum of the first five values on a line of the Pascal triangle, lines which each add up to a power of 2 .

The above argument owes its extreme elegance to its high degree of disentanglement: the first blob is not concerned with n and the second one not with f , while their combination is no more than two substitutions. Note that this partitioning into two mutually independent blobs was only possible thanks to the introduction of c and p , which carry the interface.

Acknowledgement We owe the above beautiful solution to A. Blokhuis. (End of Acknowledgement.)

The binary search

Let A be a non-empty, ascending, integer sequence of length N , more precisely, let

$N > 0$, and

$(\underline{A}\, i,j: 0 \leq i < j < N: A.i \leq A.j)$;

let X be an integer. We are requested to design a program solving the equation

present: (present $\equiv$ $(\underline{E}\, i: 0 \leq i < N: A.i = X)$) .

* * *

We distinguish two cases, $A.0 > X$ and $A.0 \leq X$; the latter case being the harder one, we deal with that one first.

It is hard to visualize an algorithm correctly establishing present, i.e. determining that the value X indeed occurs in the sequence, without determining at the same time where X occurs, i.e. without establishing for some integer, i say,

$$A.i = X \wedge 0 \leq i < N \quad .$$

As intermediate result, this is too strong to aim for: if ¬present is to be established, no such value for i exists. The best we can do in that case is to let i determine the pair of adjacent values in the sequence "between which X is missing", i.e. to establish

$$A.i < X < A.(i+1) \quad \wedge \quad 0 \leq i < N$$

for which purpose we define A.N so that

$$X < A.N \quad .$$

Combining the two intermediate results we construct the more realistic target of establishing

R: $\quad A.i \leq X < A.(i+1) \quad \wedge \quad 0 \leq i < N$.

Our program can then have the form

```
|[ var i: int
; establish R
; present:= A.i = X
]|
```

.

For "establish R", the precondition $A.0 \leq X < A.N$ suggests a repetition whose invariant P is obtained from R by replacing $(i+1)$ by a fresh variable, j say. I.e. with

P: $\quad A.i \leq X < A.j \quad \wedge \quad 0 \leq i < j \leq N$

we suggest for "establish R"

```
|[ var j: int
; i,j := 0, N {P}
; do j ≠ i+1 → shrink (j-i) {P} od {R}
]| {R}
```

.

Let us now investigate for "shrink $(j-i)$" to what extent a change of i will do the job. To this end we derive the weakest precondition such that $i:=h$ decreases $(j-i)$ under invariance of P :

$$j-h < j-i \quad \wedge$$
$$A.h \leq X < A.j \quad \wedge \quad 0 \leq h < j \leq N$$

which follows from

$$P \wedge i < h < j \quad \wedge \quad A.h \leq X \quad .$$

For $j:=h$ we derive in the same manner

$$P \wedge i < h < j \quad \wedge \quad X < A.h \quad .$$

These two preconditions lead for "shrink (j-i)" to

```
|[ var h: int
; solve h:(i<h<j)
; if A.h ≤ X → i:= h
  [] X < A.h → j:= h
  fi
]|
```

We observe, firstly, that, its guards being each other's complement, the alternative construct does not abort, and, secondly, that the precondition of shrink (j-i) – in particular: $i<j \wedge j \neq i+1$, or, equivalently, $i+1 \leq j-1$ – implies that equation $h:(i<h<j)$ has at least one solution. For reasons of efficiency we don't implement "solve $h:(i<h<j)$" by $h:= i+1$ or by $h:= j-1$, but rather by

$$h:= \text{avg}.i.j$$

The nice thing of the above derivation is that it makes it so clear that the termination of the repetition and the establishment of R are totally independent of the sequence A being ascending. Even if the algorithm establishes present we need not know that A is ascending; only the outcome ¬present requires for its trustworthiness that A is indeed ascending.

In the case $A.0>X$, $A.0>X \wedge 0\leq i<N$ is an invariant and above program establishes ¬present, as it should.

The Theorem of Pompeiu

Theorem Consider an equilateral triangle and an arbitrary point in the plane of the triangle. Then the three distances from the vertices of the triangle to the fourth point satisfy the triangular inequalities.

Proof The proof is by constructing a figure that contains a triangle with the three distances as its edge lengths. To this end we consider two instances of the triangle/point configuration, rotated over 60° around one of the vertices of the triangle.

Because the triangle is equilateral, one instance of one of the other two vertices coincides with the other instance of the third vertex. The triangle formed by this point of coincidence and the two instances of the fourth point has the desired edge lengths, as we see as follows.

Because the rotation is over 60°, the distance between the two instances of the fourth point equals their distance from the centre of rotation, which is one of the vertices. Because in the point of coincidence, different instances of the two remaining vertices coincide, its distances from the two instances of the fourth point are the distance from the fourth point to the two other vertices of the equilateral triangle. (End of Proof.)

Acknowledgement The above construction is due to G.R. Veldkamp. (End of Acknowledgement.)

Remark In my filler at the YoP Institute I used a picture, which was annoyingly overspecific: for instance, one cannot avoid choosing the fourth point inside or outside the triangle. Later in the institute, Jan L.A. van de Snepscheut showed how he could describe a class of mutual-exclusion algorithms with his hands in his pockets. The above presentation of Veldkamp's proof of the Theorem of Pompeiu has been inspired by that performance; it is a striking example of avoiding avoidable case analyses. (End of Remark.)

The monotonicity of extreme solutions

In the following, capital letters P, Q, X, Y, Z denote predicates on some space. Universal quantification over that space is denoted by surrounding the universally quantified predicate by a pair of square brackets, known as "the everywhere operator". Lower case letters f, g denote predicate transformers, i.e. functions from predicates (or predicate pairs) to predicates; functional application is denoted by an infix full stop.

Let f be monotonic in both its arguments, i.e.

$$[P \Rightarrow Q] \Rightarrow [f.P.Y \Rightarrow f.Q.Y] \wedge [f.X.P \Rightarrow f.X.Q] \quad \text{, for all } P, Q, X, Y .$$

Let, for all X, $g.X$ be the strongest solution of the equation

$$Y: [f.X.Y \Rightarrow Y] \quad ,$$

i.e.

(0) $[Z \Rightarrow f.X.(g.X)] \Rightarrow [Z \Rightarrow g.X]$ for all X, Z

(1) $[f.X.Y \Rightarrow Y] \Rightarrow [g.X \Rightarrow Y]$ for all X, Y

Remark Monotonicity of f in its second argument implies the existence of that strongest solution; (0) is equivalent to $[f.X.(g.X) \Rightarrow g.X]$, i.e. it states that $g.X$ is a solution; (1) states that $g.X$ implies each solution. (End of Remark.)

Then, g is monotonic, i.e. for all P, Q

$$[P \Rightarrow Q] \Rightarrow [g.P \Rightarrow g.Q] \quad .$$

Proof We observe for any P, Q

$$\begin{array}{ll} & [g.P \Rightarrow g.Q] \\ \Leftarrow & \quad \{(1) \text{ with } X, Y := P, g.Q\} \\ & [f.P.(g.Q) \Rightarrow g.Q] \\ \Leftarrow & \quad \{(0) \text{ with } X, Z := Q, f.P.(g.Q)\} \\ & [f.P.(g.Q) \Rightarrow f.Q.(g.Q)] \\ \Leftarrow & \quad \{f \text{ is monotonic in its 1st argument}\} \\ & [P \Rightarrow Q] \end{array} \quad .$$

(End of Proof.)

The above 3-step proof is in a sense the shortest one possible: the first step takes into account that the function applied to P is g, the second step takes into account that the function applied to Q is g, and the third step uses that f is monotonic in its 1st argument —and all three facts are indispensable—.

It has, however, been included for more than its brevity alone; it has been included because it is the simplest example of the standard derivation of proofs of theorems involving extreme solutions.

The shape of the consequent $[g.P \Rightarrow g.Q]$ immediately tells us that for the application of g to P, (0) is irrelevant, and (1) hence essential, and, conversely, that for the application of g to Q, (1) is irrelevant and, hence, (0) is essential. The above heuristics all but dictate the design of such proofs. (We point out that these heuristics would have been hardly available, had the strongest solution been defined by "it is a solution and implies all other solutions" instead of by "it is a solution and implies all solutions".)

Finally we would like to point out that the use of the follows-from symbol $\Leftarrow$ has enabled us to present the calculation without pulling a single rabbit out of a hat.

An algebraic approach to the predicate calculus

As in the previous filler, capital letters can be viewed as standing for predicates on some space, universal quantification over which is then denoted by a pair of square brackets. Because we wish to replace what is usually called "reasoning" whenever profitable by calculation, we here present the logical operators stressing their algebraic properties.

- equality between predicates X and Y is expressed by "everywhere equivalent": $[X \equiv Y]$.

- function application is characterized by the Rule of Leibniz, i.e. by being equality-preserving:

$$[X \equiv Y] \Rightarrow [f.X \equiv f.Y] \quad ;$$

expressions are postulated to be functions of their subexpressions.

- equivalence is postulated to be associative, i.e.

$$[((X \equiv Y) \equiv Z) \equiv (X \equiv (Y \equiv Z))]$$

– so that, from here on, parentheses in continued equivalences will be omitted – and to be symmetric, i.e.

$$[X \equiv Y \equiv Y \equiv X] \quad .$$

Parsing the last formula as $[X \equiv (Y \equiv Y \equiv X)]$ and as $[(X \equiv Y \equiv Y) \equiv X]$, we see that $\equiv$ has a left- and a right-identity element – viz. $Y \equiv Y$ –; therefore $\equiv$ has a unique identity element, which we denote by true :

$$[X \equiv \text{true} \equiv X] \quad .$$

- disjunction is postulated to be associative, i.e.

$$[(X \vee Y) \vee Z \equiv X \vee (Y \vee Z)]$$

– so that in continued disjunctions parentheses can be omitted –, to be symmetric, i.e.

$$[X \vee Y \equiv Y \vee X] \quad ,$$

to be idempotent, i.e.

$$[X \vee X \equiv X]$$

and to distribute over equivalence, i.e.

$$[X \vee (Y \equiv Z) \equiv X \vee Y \equiv X \vee Z] \quad .$$

o $[X \vee \text{true} \equiv \text{true}]$

Proof We observe for any X, Y

$$\begin{aligned} & X \vee \text{true} \\ = & \quad \{\text{def. of true}\} \\ & X \vee (Y \equiv Y) \\ = & \quad \{\vee \text{ distributes over } \equiv\} \\ & X \vee Y \equiv X \vee Y \\ = & \quad \{\text{def. of true}\} \\ & \text{true} \end{aligned}$$

(End of Proof.)

• conjunction is defined in terms of equivalence and disjunction by

$$[X \wedge Y \equiv X \equiv Y \equiv X \vee Y] \quad ,$$

a formula known as the Golden Rule.

o conjunction is associative, symmetric and idempotent

Proof This is left to the reader. (End of Proof.)

o $[X \wedge (Y \equiv Z) \equiv X \wedge Y \equiv X \wedge Z \equiv X]$

Proof We observe for any X, Y, Z

$$\begin{aligned} & X \wedge Y \equiv X \wedge Z \\ = & \quad \{\text{Golden Rule, twice}\} \\ & X \equiv Y \equiv X \vee Y \equiv X \equiv Z \equiv X \vee Z \\ = & \quad \{\text{rearranging terms of continued equivalence}\} \end{aligned}$$

$$
\begin{array}{ll}
 & X \equiv Y \equiv Z \equiv X \vee Y \equiv X \vee Z \equiv X \\
= & \quad \{ \vee \text{ distributes over } \equiv \} \\
 & X \equiv (Y \equiv Z) \equiv X \vee (Y \equiv Z) \equiv X \\
= & \quad \{ \text{Golden Rule} \} \\
 & X \wedge (Y \equiv Z) \equiv X \quad .
\end{array}
$$

(End of Proof.)

- $[X \wedge (U \equiv Y \equiv Z) \equiv X \wedge U \equiv X \wedge Y \equiv X \wedge Z]$,

i.e. "$\wedge$ distributes over $\equiv\equiv$" .

Proof This is left to the reader. (End of Proof.)

- $[X \vee (Y \wedge Z) \equiv (X \vee Y) \wedge (X \vee Z)]$ and
- $[X \wedge (Y \vee Z) \equiv (X \wedge Y) \vee (X \wedge Z)]$, i.e.

disjunction and conjunction distribute over each other.

Proof To prove the last one we observe for any X, Y, Z

$$
\begin{array}{ll}
 & (X \wedge Y) \vee (X \wedge Z) \\
= & \quad \{ \text{Golden Rule} \} \\
 & (X \wedge Y) \wedge (X \wedge Z) \equiv X \wedge Y \equiv X \wedge Z \\
= & \quad \{ \wedge \text{ is associative, symmetric, and idempotent} \} \\
 & X \wedge (Y \wedge Z) \equiv X \wedge Y \equiv X \wedge Z \\
= & \quad \{ \wedge \text{ distributes over } \equiv\equiv \} \\
 & X \wedge (Y \wedge Z \equiv Y \equiv Z) \\
= & \quad \{ \text{Golden Rule} \} \\
 & X \wedge (Y \vee Z) \quad .
\end{array}
$$

The preceding one can be proved similarly. (End of Proof.)

- $[X \wedge (X \vee Y) \equiv X]$ and
- $[X \vee (X \wedge Y) \equiv X]$, known as the

Laws of Absorption

Proof. To prove the first one – the second one can be proved similarly – we observe for any X, Y

$$
\begin{aligned}
& X \wedge (X \vee Y) \\
= & \quad \{\text{Golden Rule}\} \\
& X \equiv X \vee Y \equiv X \vee (X \vee Y) \\
= & \quad \{\text{associativity and idempotence of } \vee\} \\
& X \equiv X \vee Y \equiv X \vee Y \\
= & \quad \{\text{identity element of } \equiv\} \\
& X \quad .
\end{aligned}
$$

(End of Proof.)

Finally we derive for the conjunction

- $[X \wedge \text{true} \equiv X]$

Proof We observe for any X

$$
\begin{aligned}
& X \wedge \text{true} \\
= & \quad \{\text{Golden Rule}\} \\
& X \equiv \text{true} \equiv X \vee \text{true} \\
= & \quad \{\text{true is zero-element of } \vee\} \\
& X \equiv \text{true} \equiv \text{true} \\
= & \quad \{\text{identity element of } \equiv\} \\
& X \quad .
\end{aligned}
$$

(End of Proof.)

- implication is defined in terms of equivalence and disjunction by

$$[X \Rightarrow Y \equiv X \vee Y \equiv Y] \quad .$$

Using the Golden Rule the reader may derive

- $[X \Rightarrow Y \equiv X \wedge Y \equiv X]$

and using the Laws of Absorption

- $[X \Rightarrow X \vee Y]$ and
- $[X \wedge Y \Rightarrow X]$.

We shall prove

- $[X \Rightarrow (Y \Rightarrow Z) \equiv X \wedge Y \Rightarrow Z]$

Proof We observe for any X, Y, Z

$$\begin{aligned} & X \Rightarrow (Y \Rightarrow Z) \\ = & \quad \{\text{relation between } \Rightarrow \text{ and } \wedge\} \\ & X \wedge (Y \wedge Z \equiv Y) \equiv X \\ = & \quad \{\text{relation between } \wedge \text{ and } \equiv\} \\ & X \wedge Y \wedge Z \equiv X \wedge Y \\ = & \quad \{\text{relation between } \Rightarrow \text{ and } \wedge\} \\ & X \wedge Y \Rightarrow Z \quad . \end{aligned}$$

(End of Proof)

- $[X \wedge (X \Rightarrow Y) \equiv X \wedge Y]$

Proof We observe for any X, Y

$$\begin{aligned} & X \wedge (X \Rightarrow Y) \\ = & \quad \{\text{relation between } \Rightarrow \text{ and } \wedge\} \\ & X \wedge (X \wedge Y \equiv X) \\ = & \quad \{\text{relation between } \wedge \text{ and } \equiv\} \\ & X \wedge X \wedge Y \equiv X \wedge X \equiv X \end{aligned}$$

$=$ {idempotence of $\wedge$}
$X \wedge Y \equiv X \equiv X$
$=$ {identity element of $\equiv$}
$X \wedge Y$.

(End of Proof.)

Implication is transitive, i.e.

○ $[(X \Rightarrow Y) \wedge (Y \Rightarrow Z) \Rightarrow (X \Rightarrow Z)]$.

Proof We observe for any X, Y, Z

$[(X \Rightarrow Y) \wedge (Y \Rightarrow Z) \Rightarrow (X \Rightarrow Z)]$
$=$ {preprevious theorem}
$[X \wedge (X \Rightarrow Y) \wedge (Y \Rightarrow Z) \Rightarrow Z]$
$=$ {previous theorem}
$[X \wedge Y \wedge (Y \Rightarrow Z) \Rightarrow Z]$
$=$ {previous theorem}
$[X \wedge Y \wedge Z \Rightarrow Z]$
$=$ {Law of Absorption in implicative version}
true .

(End of Proof.)

○ $[(X \Rightarrow Y) \vee (Y \Rightarrow Z)]$

Proof We observe for any X, Y, Z

$[(X \Rightarrow Y) \vee (Y \Rightarrow Z)]$
$=$ {relation between $\Rightarrow$ and $\vee$}
$[(X \vee Y \equiv Y) \vee (Y \vee Z \equiv Z)]$
$=$ {$\vee$ distributes over $\equiv$}
$[X \vee Y \vee Y \vee Z \equiv X \vee Y \vee Z \equiv Y \vee Y \vee Z \equiv Y \vee Z]$
$=$ {idempotence of $\vee$; identity element of $\equiv$}
true .

(End of Proof)

○ $[(X \Rightarrow Y) \wedge (Y \Rightarrow X) \equiv X \equiv Y]$

Proof We observe for any X, Y

$(X \Rightarrow Y) \wedge (Y \Rightarrow X)$
$=$ {Golden Rule; previous theorem}
$X \Rightarrow Y \equiv Y \Rightarrow X$
$=$ {definition of $\Rightarrow$}
$X \vee Y \equiv Y \equiv X \vee Y \equiv X$
$=$ {identity element of $\equiv$}
$X \equiv Y$.

(End of Proof.)

So much for the implication. Note that conjunction and implication have been defined in terms of equivalence and disjunction. In order to introduce the negation we therefore postulate its properties with respect to the latter two connectives only; its properties with respect to conjunction and implication can then be derived.

● negation and equivalence are postulated to be connected by

$[\neg(X \equiv Y) \equiv \neg X \equiv Y]$.

○ $[\neg X \equiv Y \equiv X \equiv \neg Y]$

Proof We observe for any X, Y

$\neg X \equiv Y$
$=$ {connection between $\neg$ and $\equiv$}
$\neg(X \equiv Y)$
$=$ {connection between $\neg$ and $\equiv$; symmetry of $\equiv$}
$X \equiv \neg Y$.

(End of Proof.)

Substitution $Y := \neg X$ in the above yields

○ $[X \equiv \neg\neg X]$.

• negation and disjunction are postulated to be connected by the Law of the Excluded Middle, i.e.

$[X \vee \neg X]$.

Exploring what we can derive from the above two postulates for the negation, we observe for any X, Y

true
= { Excluded Middle , $X := X \equiv Y$ }
$[(X \equiv Y) \vee \neg(X \equiv Y)]$
= { relation between $\neg$ and $\equiv$ }
$[(X \equiv Y) \vee (\neg X \equiv Y)]$
= { $\vee$ distributes over $\equiv$ }
$[X \vee \neg X \equiv X \vee Y \equiv Y \vee \neg X \equiv Y \vee Y]$
= { Excluded Middle; idempotence of $\vee$ }
○ $[\neg X \vee Y \equiv X \vee Y \equiv Y]$

Confronting the above theorem with the definition of the implication we get

○ $[X \Rightarrow Y \equiv \neg X \vee Y]$.

We can furthermore use it to derive the Laws of de Morgan:

○ $[\neg X \vee \neg Y \equiv \neg(X \wedge Y)]$

○ $[\neg X \wedge \neg Y \equiv \neg(X \vee Y)]$.

Proof We shall prove the first one. To this end we observe for any X, Y

$$
\begin{array}{ll}
 & \neg X \vee \neg Y \\
= & \quad \{\text{recent theorem with } Y := \neg Y\} \\
 & X \vee \neg Y \equiv \neg Y \\
= & \quad \{\text{same theorem with } X, Y := Y, X\} \\
 & X \vee Y \equiv X \equiv \neg Y \\
= & \quad \{\text{relation between } \neg \text{ and } \equiv\} \\
 & \neg (X \vee Y \equiv X \equiv Y) \\
= & \quad \{\text{Golden Rule}\} \\
 & \neg (X \wedge Y) \ .
\end{array}
$$

(End of Proof.)

We leave to the reader the derivation of the analogous

$$\circ \quad [\neg X \wedge Y \equiv X \wedge Y \equiv \neg Y] \ .$$

Etc.

* * *

The above text is somewhat more elaborate than what I showed as filler at the YoP Institute; also here the manipulation of formulae with explicit quantification has been omitted. The material has been included because it deserves to be better known than it is. We regret that classical logic has not yet become a daily, calculational tool of the working mathematician, for we find it indispensable.

The reason why logic is so little used is probably that it has been presented in the wrong way, viz. as formalization of how mathematicians "think" instead of a calculus in its own right. As the

reader will have noticed, we have allowed the equivalence to play a very central rôle. For a calculational approach, this is essential: the notion of function application is defined by the fact that it is equality-preserving. And it is precisely the equivalence that is the logical connective that is the hardest to render in natural language; we have the "if and only if", but its deficiencies are clearly displayed in the following sentence: "John sees with both eyes if and only if John sees with one eye if and only if John is blind.". By all linguistic standards, this sentence is total gibberish.

The purpose of logic is not to mimic verbal reasoning but to provide a calculational alternative.

Nuenen, 21 December 1987

prof. dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78712-1188
United States of America