

Kenneth Iverson was a mathematician who is most famous for designing APL. This was the name of his programming language, and it cleverly stood for “A Programming Language.” The language is unique—unlike almost any other language—and contains many powerful and interesting ideas. He won the 1979 Turing Award for this and related work.

We will talk about notation in mathematics and theory, and how notation can play a role in our thinking.

When I was a junior faculty member at Yale University, in the early 1970s, APL was the language we used in our beginning programming class. The reason we used it was simple: Alan Perlis, the leader of the department, loved the language. I was never completely sure why Alan loved it, but he did. And so we used it to teach our beginning students how to program.

Iverson had created his language first as a notation for describing complex digital systems. The notation was so powerful that even a complex object like a processor could be modeled in his language in relatively few lines of code. The lines might be close to unreadable, but they were few. Later the language was implemented and had a small but strong set of believers. Clearly Alan was one of them, since he once said:

A language that doesn’t affect the way you think about programming is not worth knowing.

APL was great for working with vectors, matrices, and even higher-order objects. It had an almost uncountable number of built-in symbols that could do powerful operations on these objects. For example, a program to find all the primes below R is:

$$(\sim R \in R \circ . \times R) / R \leftarrow 1 \downarrow \iota R.$$

This is typical APL: very succinct, lots of powerful operators, and no keywords. The explanation of this program is given by Brad McCormick on a wonderful page about APL.

One comment about APL is:

[The language was] famous for its enormous character set, and for being able to write whole accounting packages or air traffic control systems with a few incomprehensible key strokes.

Albeit not quite right and not quite nice, this comment does capture the spirit of Iverson’s creation. To make complex functions expressible tersely was an interesting idea. For example, Michael Gertelman has [written](#) Conway’s Game of Life as one line of APL:

$$\begin{aligned} \Phi' \square', \in N\rho \subset S \leftarrow' \leftarrow \square \leftarrow (3 = T) \vee M \wedge 2 \\ = T \leftarrow \supset + / (V\phi'' \subset M), (V\Theta'' \subset M), (V, \phi V)\phi''(V, V \leftarrow 1^-1)\Theta'' \subset M' \end{aligned}$$

This should make it clear that this language was powerful, perhaps too powerful.

When I had to learn APL in order to teach the beginning class, I decided to start a project on building a better implementation. The standard implementation of the language was as an interpreter, while one of my Yale PhD students, Tim Budd, eventually wrote a compiler for APL. We also proved in modern terms a theorem about the one-liners of the language:

Theorem 2.1 *Every APL one-line program can be computed in logspace.*

Proving this was not completely trivial, since the operators are so powerful. Perhaps another time I will discuss this work in more detail. Tim wrote a [book](#) on his compiler.

Let’s turn to discuss various notations used in math and theory.

2.1 Good Notation

It is hard to imagine that there was a time when mathematicians did not even have basic symbols to express their ideas. For the many who believe that notation helps to shape the way we think, clearly without basic symbols modern mathematics would be impossible—or at least extremely difficult.

- **Robert Recorde** is credited with introducing the equality symbol in 1557. He [said](#)

“... [T]o avoid the tedious repetition of these words, ‘is equal to,’ I will set (as I do often in work use) a pair of parallels of one length (thus =), because no two things can be more equal.

- **René Descartes** is known for the first use of superscripts to denote powers:

$$x^4 = x \cdot x \cdot x \cdot x.$$

François Viète introduced the idea of using vowels for unknowns and consonants for known quantities. Descartes changed this to: use letters at the end of the alphabet for unknowns and letters at the beginning for knowns.

One amusing part of this notation was that Descartes thought that x, y, z would be equally used by mathematicians. But it is x this and x that... The story—maybe a legend only—is that there is a reason that x became the dominant letter to denote an unknown. Printers had to set Descartes’ papers in type,

and they used many y 's and z 's, since the French language uses them quite a bit. But French almost never uses x , so Descartes' *La Géométrie* used x as the variable most of the time.

- **Isaac Newton and Gottfried Leibniz** invented the calculus. There is a controversy that continues to this day on who invented what and who invented what first. This is sometimes called the “[calculus war](#).”

Independent of who invented what, they did use different notations, at least that is without controversy. Newton used the dot notation and Leibniz the differential notation. Thus Newton would write \dot{x} while Leibniz would write $\frac{dx}{dt}$ for the same thing. The clear winner, most agree, is the better notation of Leibniz. It is even claimed that the British mathematicians by sticking to Newton's poorer notation lagged behind the rest of Europe for decades.

- **Leonhard Euler** introduced many of the common symbols and notations we still use today. He used e and π for the famous constants and i for the square root of -1 . For summation Euler used Σ and also introduced the notation for functions: $f(x)$. One can look at some of his old papers and see equations and expressions that look quite modern—of course they are old, but look modern because we still use his notation.
- **Carl Gauss** introduced many ideas and proved many great theorems, but one of his most important contributions concerns the congruence notion. Leonhard Euler had earlier introduced the notion, but without Gauss' brilliant [notation](#) for congruences, they would not be so easy to work with. Writing

$$x \equiv y \bmod m$$

is just magic. It looks like an equation, can be manipulated like an equation—well almost—and is an immensely powerful notation.

- **Paul Dirac** introduced in 1939 his famous [notation](#) for vectors. His so-called *bra-ket* notation is used in quantum everything. It is a neat notation that takes some getting used to, but seems very powerful. I wonder why it is not used all through linear algebra. A simple example is:

$$|x\rangle = [a_0, a_1, \dots, a_n]^T.$$

The power of the notation is that x can be a symbol, expression, or even words that describe the value of a quantum state.

For a neater example, the outer-product of a vector R with itself, as used in the APL program for primes above, is written this way in Dirac notation:

$$|R\rangle\langle R|,$$

which flips around the inner product $\langle R|R\rangle$. Now to multiply the matrix formed by the outer product by a row vector a on the left and a column vector b on the right, we write

$$\langle a|R\rangle\langle R|b\rangle.$$

The bra-kets then associate to reveal that this is the same as multiplying two inner products. Another neat feature is that $\langle x|$ versus $|x\rangle$ captures the notion of a *dual vector*, and when x has complex entries, the $\langle x|$ notation implicitly complex-conjugates them.

- **Albert Einstein** introduced a notation to make his General Relativity equations more succinct. I have never used the notation, so I hope I can get it right. According to his rule, when an index variable appears twice in a single term, once as a superscript and once as a subscript, then it implies a summation over all possible values. For instance,

$$y = c_i x^i$$

is

$$y = \sum_{i=1}^3 c_i x^{(i)},$$

where $x^{(i)}$ are not powers but objects.

- **Dick Karp** introduced the notation $C/f(n)$ to complexity theory in our joint paper on the Karp–Lipton Theorem. Even though the notation was his invention, as his co-author I will take some ϵ amount of credit.

2.2 Good Notation?

Not all notation that we use is great; some may even be called “bad.” I would prefer to call these good with a question mark. Perhaps the power of notation is up to each individual to decide. In any event here are a few “good?” notations.

- **John von Neumann** was one of the great mathematicians of the last century, and helped invent the modern computer. He once introduced the notion

$$f(((x)))$$

where the number of parentheses modified the function f . It does not matter what they denoted, the notation could only be used by a brilliant mind like von Neumann’s. This notation is long gone as used by von Neumann, but in the theory of ideals, $((p))$ is different from (p) . Oh well.

- The letter π for pi is fine, but maybe it denotes the wrong number? Since 2π denotes the full unit circle and occurs all the time in physics, maybe we should have used the symbol π for that number? See a further discussion by Lance Fortnow and Bill Gasarch referenced in the end notes.
- Why is the charge of the electron negative? Evidently it is because Benjamin Franklin believed the flow of an unseen fluid was opposite to the direction in which the electron particles were actually going.
- Why has humanity been unable to establish that \subset means proper subset and only \subseteq means subset, by analogy to $<$ and \leq ? Hence for proper subset one often resorts to the inelegant \subsetneq notation.

- I will end with one example of notation that many feel strongly is only “good?”: $f'(x)$ to denote the derivative of a function. This generated a lively [discussion](#) fifteen years ago on the long-running Drexel University Math Forum, which is also referenced in the end notes.

2.3 Open Problems

Does good notation help make mathematics easier? Are there some notions that are in need of some better notation? What do you think?

2.4 Notes and Links

Original post:

<http://rjlipson.wordpress.com/2010/11/30/notation-and-thinking/>

Brad McCormick’s APL page:

<http://www.users.cloud9.net/~bradmcc/APL.html>

Quotation on APL:

www-users.cs.york.ac.uk/~susan/cyc/p/prog.htm

Conway’s “Life” in APL:

<http://catpad.net/michael/apl/>

Book by Timothy Budd:

<http://web.engr.oregonstate.edu/~budd/Books/aplc/>

Equals sign reference:

http://en.wikipedia.org/wiki/Equals_sign

La Géométrie:

http://en.wikipedia.org/wiki/La_Geometrie

Leibniz–Newton controversy:

http://en.wikipedia.org/wiki/Leibniz_and_Newton_calculus_controversy

Gauss congruence notation:

http://en.wikipedia.org/wiki/Modular_arithmetic#Congruence_relation

Dirac bra-ket notation:

http://en.wikipedia.org/wiki/Bra-ket_notation

Einstein summation notation:

http://en.wikipedia.org/wiki/Einstein_notation

Pi discussion by Fortnow and Gasarch:

<http://blog.computationalcomplexity.org/2007/08/is-pi-defined-in-best-way.html>

Drexel Math Forum discussion on notation for derivatives:

<http://mathforum.org/kb/thread.jspa?threadID=33142&messageID=108756>

This post in *Gödel’s Lost Letter* had an exceedingly lively comment discussion.

Picture credits:

1. Conway’s Life: Screen capture from <http://catpad.net/michael/apl/>

<http://www.springer.com/978-3-642-41421-3>

People, Problems, and Proofs

Essays from Gödel's Lost Letter: 2010

Lipton, R.J.; Regan, K.W.

2013, XVIII, 333 p. 33 illus., 5 illus. in color., Hardcover

ISBN: 978-3-642-41421-3