acmqueue

Two Books Alike in Dignity

Formal and informal approaches to C++ mastery

Stan Kelly-Bootle

Woke up this morning ... surprised to find my Sennheisers still connecting ears to my new MacBook Pro, with iTunes set to blues genre in shuffle mode. Lest you think I've succumbed to the despicable placement temptation that seduces so many columnists and filmmakers in these pursy times,¹ I'm reluctant to price the named products, plug their sources (Amazon, eBay, Marple, etc.), or elaborate on the immense pleasure I derive from their splendid cost-effective performances. Suffice it to mention that the subliminal album playing all night was *Broke, Black and Blue, Volume 1*, available for 7.95 US dollars or 7.95 Apple pounds sterling.² The album title (stretching the meaning of *download* to *down-and-out load*) is also remarkably appropriate for those living in Iceland, Greece, Ireland, and other economies suffering the effects of bankers' generosity fueling consumer greed.³

Last year was a mixed year in my bookish life. No shortage of excellent tomes but, alas, too many months of medically enforced prostration, conducive to reading rather than rent-paying writing. I should also add book listening, since audiobooks have proved a blessing when cruel Nurse Cavil orders, "Lights out! Extinguish all smoking materials!!" I have this great deal with audible.com whereby I get two FREE downloads each month from its infinite catalogue. I use the term FREE in the gullible gambler's sense. I do pay a modest retainer giving me the monthly challenge of selecting two titles that list at more than the fixed subscription, whether I like them or not. I'm joking, of course. The expensive titles usually match my natural preferences. I accidentally discovered last month that audible.com employs a remarkably customer-friendly algorithm. I had thoughtlessly ticked three titles: a @ £x (free); b @ £y (free); c @ £z (buy). Ignoring the P=NP controversy, audible.com's computer immediately spotted that $x > z > y$, and revamped my order to: a @ £x (free); b @ £y (buy); c @ £z (free). I intend to test their algorithm with more complex orders, aiming at aleph-0 titles, or as close as time allows.

Incidentally, there's an emerging meta-controversy: can Vinay Deolalikar's claimed disproof of P=NP be verified in polynomial time by an exponentially increasing number of qualified online doubters? We can now add proof by outgroup blog to the usual list of Popperian suspects (proof by induction; proof by repeated assertion; proof by authority-other-than-Popper's, ...).

Thus, for my relistening pleasure I now have eight hours of Nabokov's short stories and 15 hours of Richard Feynman LIVE from his Cal Tech podium, digitally remastered and proving that Nooten is an acceptable pronunciation for Newton if you happen to be a genius from da Bronx.

There are downsides to mathematical physics on audiobooks when the narrator is equation-deprived and notation-immune. Older fans may recall my *Unix Review* Devil's Advocate columns where soundo was introduced as the spoken analogy for typo. Annoyingly funny examples from recent audiobooks include Einstein's *Relativity: The Special and the General Theory*. It's convincingly, but quite unnecessarily, narrated in a thick Germanic accent, yet completely garbles all the formulae. Primed coordinates are pronounced "x to the one," and t-squared becomes "t to the two" (quick pause to ponder if Einstein

was familiar with “Tea for Two” from the 1925 musical, *No, No Nannette?*). Less obvious to the lay listener is a recurring sound in the otherwise delightful narration of John Polkinghorne’s *Quantum Theory: A Very Short Introduction*. The Greek letter nu (ν), the standard symbol for frequency, comes out as “vee.” Many books, then, regularly swim into my ken. Ken then ranks them by readability, and passes a selection to me, depending on my perceived state of health. As you may know, after sampling the recumbent pleasures of cupuration, I’ve taken up permanent recuperation. This has a Viconian, cyclic ring to it, recalling an indolent, impoverished Scouse youthhood. I would counter my mother’s attempts to rouse me at the crack of noon, with a croaked “I’ll gerra job termorra, mam.” (rough translation: “Fear not, mater, I plan to seek employment in the near future.”).⁴

One book that Ken withheld until he felt I could bear the shock turned out to be the worst book ever on computer history.⁵ It suffers from being relentlessly set in the semi-legible E13B font found on bank checks (whence the expression, “Our bottom line leaves much to be desired.”). Even my document scanner has trouble reading Taschen’s version of a font intended for mechanical rather than human eyes. One reason for this is the random choice of foreground and background colors, breaking every axiom of typographical legibility. The unpardonable sin, though, is that the Cambridge EDSAC gets no mention at all. I don’t begrudge the ample patriotic praise for Konrad Zuse, but... this omission reached me last December as we heard of the death of Sir Maurice Wilkes, EDSAC’s *onlie begetter*. (The obituary I wrote for him is included here.)

Returning to my columnar title, I offer an explanatory song, a self-parody of my Shakespearean parody, “Romeo & Juliet”:⁶

Two books alike in dignity,
In C++ we set our scene,
Where Bjarne takes a gentle path
While Stepanov is hard and mean.
(Stan Kelly © Universal Music Publishing Co.)

The two books are:

- *Programming: Principles and Practice Using C++*, Bjarne Stroustrup, Addison-Wesley, 2009.
- *The Elements of Programming*, Alexander Stepanov and Paul McJones, Addison-Wesley, 2009.

I’ll refer to the books as *PPPC++* and *EoP*, and to the authors as BS and AS/PM. It’s clear from first sight, title, and *bulk* that the books are taking different paths to enlightenment! *PPPC++* is 1,236 pages; *EoP* is 262 pages.

A useful but oversimplified analogy is between *EoP* as *formal* and *PPPC++* as *informal* mathematics. In a review of *Proof and Other Dilemmas: Mathematics and Philosophy*, Ethan Akin amusingly highlights the distinction between the two concepts:⁷

“As with *certainty*, the word *proof* is used for two different things, and it is worth making the distinction explicit. Fundamental is logical proof, hereafter called *PROOF*. Happily this is completely formalized by modern logic, so we can be absolutely clear about what it is: a sequence of statements each of which is either (1) an axiom, a definition, or a previously *PROOFed* result or (2) a statement obtained from the earlier ones on the list by one of the (purely formal) rules of inference ... It is the mechanical and almost mind-independent character of this idea of *PROOF* which justifies the hope

(if we can call it that) of the Hilbert Program. Furthermore, when we are assailed by the barbarians (i.e., philosophers!) PROOF is the citadel to which we retreat.”

Here we see the common suspicion of over-philosophizing by “real work-a-day” mathematicians, which echoes the practical quantum mechanics’ quip: “Shut up and keep calculating.”

Remembering Sir Maurice Wilkes

(Reproduced with permission from The Ring: The Journal of the Cambridge Computer Lab Ring, Issue XXVI, January 2011.)

I was one of the fortunate first trio of Cambridge mathematics graduates to benefit from Maurice Wilkes’ pioneering genius, not only in launching the EDSAC, but also in devising the world’s first postgraduate computer science diploma. Indeed, back in 1953 the term *computer science* had not yet emerged, and the course suffered from the rather arcane mouthful, “Diploma in Numerical Analysis and Automatic Computing.”

The emphasis on numerical analysis was important. These days, many computer users will accept whatever numbers are churned out by their PCs without prior concern for estimated errors. We were equipped with heavy metal, multibarreled Brunsviga calculators, trained to be aware of what our sums were doing—close up—before we ran the program (or programme as it then was).

No obituary can capture the shock his students feel at his departure. His longevity somehow seemed inviolate as he continued so actively year after year, outliving whole generations of EDSAC alumni.

Nor is it easy to do justice to Maurice Wilkes, the man. He had a wry, restrained sense of humor that can only be described, inadequately, as English. It is present in his memoirs and even in his technical writings. Happily it also survives in videos of EDSAC reunions and anniversary celebrations. One example from many: he relates the story of an early EDSAC demonstration arranged at short notice. The program chosen to dazzle the nontechnical ministerial visitors was a random number generator, meaning, of course, that judging its accuracy and performance was far from trivial even for experts. Added insurance came from a generous wine service.

As the years rolled by, and the number of EDSAC alumni multiplied on the faces of the world’s best campuses and IT corporations, Maurice Wilkes managed wonderfully to keep in touch with the diaspora, a veritable who’s who of computerdom. In the late ’70s, some 30 of us welcomed him to a reunion in Silicon Valley organized by Unix guru and EDSAC graduate Steve Bourne. The tradition continues with regular reunions of members of the Cambridge Computer Lab Ring via its own magazine, *The Ring*.

Parting glasses are being lifted worldwide. Bjarne Stroustrup (inventor of C++ and yet another of Wilkes’ notable protégés) interrupted his lecture at Texas A&M University. His theme was “Giants walked in the world, then!”

The ’50s were strange and heroic times at the Maths Lab, coaxing the huge, temperamental beast to produce results without any of the mollycoddling high-level programming aids available today. The main “fast” memory was ultrasonic waves traversing tanks of mercury. We had only 1 K (1,024) 18-bit words; at the time, that seemed plenty. Wilkes and his fine team, including David Wheeler and Sandy Douglas, worked miracles in squeezing performance from EDSAC’s limited hardware, most notably by inventing the “floating” subroutine, whereby libraries of complex programs on prepunched paper tapes could be spliced into our own program tapes.

Akin goes on to proclaim the Mathematicians' Creed: briefly, that the lowercase *proof* of theorems signifies a belief that behind the less formal statement-sequences we find in the journals, there is a PROOF that can, in theory, be constructed for any nitpicking doubters.

EoP does an intriguing job of axiomatizing the foundations of programming. We are led from the Categories of Ideas: Entity, Species, Genus, through to the accumulative definitions (à la Backus-Naur) of Values, Objects, Procedures, and Types. Later chapters cover Transformations and their Orbits, Associative Operations, and Linear Orderings, ending with Composite Objects. An appendix by Sean Parent and Bjarne Stroustrup (assuring us that he supports the AS/PM deductive approach) defines the subset of C++ used in *EoP* to put meat on the axiomatic bones, as it were.

When we turn from *EoP*'s PROOFS to *PPPC++*'s proofs, we see at once that BS targets a wider audience, including those who would certainly find *EoP* forbiddingly abstract. So, what's new? It has been a constant challenge since computer programming emerged as a vital subject on the world's curricula: whom to teach what and how? It has been my privilege over the years to meet successful programmers from amazingly diverse backgrounds. One would have thought that a firm grounding in mathematical logic was essential (what do IF and THEN mean!), yet many profitable commercial computer projects are the result of *creatively* bolting together existing modules with the minimum of computer-scientific know-how.

I see as the market for *EoP* those needed to write the modules, as well as the many mathematicians, engineers, and scientists with the requisite powers of abstraction. *PPPC++* is not without its hard, challenging bits (C++ *is* a complex language), but, dare I say, it's entertaining, with many historical diversions. I hope this brief review guides your choice. My inevitable advice is: buy them both. Q

REFERENCES

1. It's often difficult to distinguish newspaper editorials from advertisements set adjacently in identical fonts. A concession from the more honest journals is a warning in nano-condensed-gothic: "This article was funded by Microsoft." Actors seen driving cars, smoking cigars, wearing shirts, or pretending to operate computers are usually positioned to reveal their good tastes and favorite brands—unless, of course, the plot calls for equipment malfunction, in which case all labels are removed to avoid litigation. I encountered a blatant example of product placement in the 1960s when I was a regular guest on the Eamonn Andrews TV show. On-camera smoking was quite common at the time, but Eamonn left no doubt as to his preferred puff: a 10-pack carton of Pl***rs cigarettes was prominently visible throughout the program.
2. One can defend Apple's UK unitary exchange-rate strategy on the grounds of arithmetical simplicity. I've no idea if similar one-to-one conversions apply to the Zimbabwe ZWD dollar or North Korean KPW won.
3. Booming literature on the global financial meltdown (2007 - ...) continues to finger the usual scapegoats. See Michael Lewis's *The Big Short*; Terry Eagleton's *The Meaning of Life—A Very Short Introduction*; *The Book of Job*; and my own forthcoming *The Mottled Swan*. One counterintuitive point made by Lewis is that the earliest signs of mortgage-industry troubles concerned the problem of early loan repayment (and loss of interest via frequent remortgaging) rather than the subsequent spate of nonpayments of overgenerous subprime loans. Economics remains the dismal science, even more so given the miserable faces of those whose Nobel-winning equations proved

so unhelpful in the wrong hands. Don Arley's *Predictably Irrational* offers further insights into the limits of formal games theory in a marketplace peopled by gullible gamblers all using the same software.

4. Speigl, F., Shaw, F., Kelly-Bootle, S. 1966. *Lern Yerself Scouse—How to Talk Proper on Merseyside*. Liverpool: Scouse Press.
5. Wurster, C. 2002. *Computers—An Illustrated History*. Taschen.
6. Available (for a song) from Apple's fine iTunes library. Search for the album, *Some People Are On the Pitch—Songs from Football's Archives*; tracks by Stan Kelly: "Romeo & Juliet"; "Wives Take This Warning." I also acknowledge some inspiration from the Bernstein/Sondheim musical, *West Side Story*. My song has a happy ending: a parody squared, as it were, proving the mathematically satisfying equation, tragedy squared = comedy.
7. Akin, E. 2010. Reviews: Proof and other dilemmas: mathematics and philosophy. *The American Mathematical Monthly* 117 (10).

LOVE IT, HATE IT? LET US KNOW

feedback@queue.acm.org

STAN KELLY-BOOTLE (<http://www.feniks.com/skb/>; <http://www.sarcheck.com>), born in Liverpool, England, read pure mathematics at Cambridge in the 1950s before tackling the impurities of computer science on the pioneering EDSAC I. His many books include *The Devil's DP Dictionary* (McGraw-Hill, 1981), *Understanding Unix* (Sybex, 1994), and the recent e-book *Computer Language—The Stan Kelly-Bootle Reader*. *Software Development Magazine* has named him as the first recipient of the new annual Stan Kelly-Bootle Eclectech Award for his "lifetime achievements in technology and letters." Neither Nobel nor Turing achieved such prized eponymous recognition. Under his nom-de-folk, Stan Kelly, he has enjoyed a parallel career as a singer and songwriter. He can be reached at curmudgeon@acmqueue.com.

© 2011 ACM 1542-7730/11/0200 \$10.00