Matter for Debate
"DIYMathematics" v. "ThePackage"

JOHN D. MAHONY

The use of mathematical software packages to obtain numerical outputs to problems in the physical sciences is a very well established and useful practice.

But what happens when the software needed is not accessible for some reason or too expensive? The alternative may be to use a spreadsheet provided by one's existing software or to develop one's own programs. This is a do-it-yourself approach that should be valuable for a sixth form student of mathematics or aspiring undergraduate who might well end up using commercial software packages in an engineering environment but who for the time being would benefit from building a suite of programs of their own.

Scientific calculators and computers normally have built-in routines for commonplace operations such as evaluating trigonometric functions or curve-plotting. These routines are building blocks for further understanding but here we are discussing the sophisticated software which sometimes hinders an understanding of the fundamentals in a discipline.

Standard packages can have other shortcomings. For example, they may encourage us to take as infallible what others have done rather to be responsible for our own work.

Documentation in user manuals is not always helpful, particularly in some fields of engineering, underlying assumptions not always being spelled out in detail. Engineers and physicists can be happy to use such software until the results do not match up in practice. Then it is left to the mathematician to fathom reasons for the disparities. This author encountered just such a scenario when working in an industry that relied on packages with little in the way of caveats regarding applicability.

There are numerous situations where it is acceptable to employ sophisticated packages, such as the algorithms in the NAG (National Algorithms Group) library. Use of these occurs usually when the practitioner is involved in high-flying Research and Development activities. However, as new computer languages arise it becomes necessary to rewrite the old algorithms and constant updates can be a source of frustration.

There are examples in the literature of software packages which serve as useful case studies for a teacher of mathematics or computer science who is keen to encourage students to develop software either on their own or communally in a class setting. This can reinforce an understanding of the fundamentals in a science, leading to confidence and self-reliance. The author would be happy to provide the interested reader with an instance showing how DIY software can be developed to produce results that compare well with those from a commercially available package.

https://doi.org/10.1017/mag.2019.15 Published online by Cambridge University Press

It is important that mathematicians starting out in STEM disciplines are encouraged not to lean too heavily on software before they understand just what underpins it. When the time comes for them to climb the corporate ladder and move into management positions, they may leave a software development environment. However, there will be those who remain engaged in mathematical activity and may at times be working away from their base, on-site for a client. They will have to rely either on industry-standard, benchmarked software that is bought in or on in-house company software. In either case they should be encouraged to check outputs where possible with results from software they have developed personally. It has been the author's experience to have faced many times – sometimes in a teamwork setting – the prospect of reconciling results from packages with results from one's DIY archive. The process is symbiotic in that one may be able to improve one's own software, and sometimes even the package, if the code is available.

Caveat emptor is as relevant today in addressing issues in software as it has ever been, irrespective of the origins of the package, be it commercial or simply company-confidential.

10.1017/mag.2019.15
JOHN D. MAHONY
5 Bluewater View, Mt. Pleasant, Christchurch 8081, New Zealand
e-mail: *johndmahony@gmail.com*

Sir Michael Atiyah (1929–2019)

It is with great sadness that The Mathematical Association has learnt of the death of Sir Michael Atiyah at the age of 89. Sir Michael was one of the great luminaries of British mathematics in recent decades, with major contributions to algebraic geometry, topological K-theory, index theory and gauge theory.

He held a string of top academic positions, including the Savilian Professor of Geometry at Oxford University, Director of the Isaac Newton Institute for Mathematical Sciences and Master of Trinity College, Cambridge. Sir Michael also held the Presidency of the Royal Society of London and of the Royal Society of Edinburgh.

He was awarded the Fields Medal in 1966 and the Abel Prize in 2004. In recognition of these achievements, he was knighted in 1983 and became a member of the Order of Merit in 1992.

Sir Michael served as MA President in 1981-82 and was elected as an Honorary Member of the Association in 2016.

For the *Gazette*, he wrote two papers. The first was 'What Is Geometry? The 1982 Presidential Address' **66** (October 1982) pp. 179-184 and the second 'Geometry and Physics' **80** Centenary Issue (March 1996) pp. 78-82.

We hope to publish a full obituary in the July *Gazette*.