Subject: Re: Numerical Recipes Article
Posted by Paul E Howland on Mon, 10 Nov 1997 08:00:00 GMT
View Forum Message <> Reply to Message

David Foster wrote:

> You'll have to pardon me, but I'm not a Mathematica user, and the
> code here looks like it was scraped off the walls of some Egyptian
> temple. If you were to show the IDL code to a programmer not familiar
> with IDL, he/she could probably figure out what it's doing. Show
> the Mathematica code to a programmer not familiar with Mathematica
> and he'll probably think your type-writer broke.
>
> There's often a trade-off between elegance/simplicity and
> functionality. Is Mathematica's sorting capabilities that much more
> flexible and powerful to justify such strange syntax?

Mathematica's programming capabilities are considerably more flexible and powerful than those of IDL, although on a simple problem like the example in the "Numerical Recipes" article they are not revealed. IDL code, however, runs much faster. Hence there is a trade off between programming time and execution time: which leads back to my original statement that you should use the most appropriate tool for the job.

I agree that Mathematica's code can look quite odd to those don't program it, but I don't regard this as a problem. I'm sure my Mum would have a better chance of understanding COBOL than IDL or Mathematica, but that doesn't mean that we should all start using COBOL! As long as an IDL programmer can understand IDL code, and a Mathematica programmer can understand Mathematica code, that's all that matters. Incidently, it took me about the same length of time to figure out what both the Mathematica and IDL code examples were doing, in the original article.

Mathematica supports a number of programming paradigms, including pattern matching, list processing, pure functions (lambda calculus), matrix/vector operations, recursive programming, symbolic mathematics, function overloading, etc. as well as the traditional DO-loop approach, and hence its code can seem quite complex. It is not well suited to array based number crunching exercises though, unlike IDL, which is optimised for this.

IDL is great. Mathematica is great. My only problem is with those who attempt to rank software tools on the basis of a single test, particularly when that test appears to have been devised to suit a particular product. It is not particularly helpful for anyone.

Paul

Paul E Howland PhD MEng CEng MIEE Room BY209
Senior Scientist DERA (Malvern)
Land Systems Sector St Andrews Road
Defence Evaluation & Research Agency Malvern
tel. +44-(0)1684-895767 Worcestershire
fax. +44-(0)1684-896315 UK

Email PEHowland@dera.gov.uk
Web Site http://www.dera.gov.uk