

## Re: A question on Newton's Method

---

*Source:* <http://sci.tech-archive.net/Archive/sci.math.num-analysis/2005-04/msg00020.html>

---

- *From:* Jon Harrop <[usenet@xxxxxxxxxxxxxxxx](mailto:usenet@xxxxxxxxxxxxxxxx)>
  - *Date:* Sat, 02 Apr 2005 06:09:01 +0100
- 

beliavsky@xxxxxxx wrote:

> Jon Harrop wrote:  
>> I can't believe you'd advise a noob to use a compiler which is still  
> getting  
>> 5 bug reports per day. How can you sleep at night?  
>  
> G95 compiles and runs the codes listed at  
> [http://www.g95.org/g95\\_status.html](http://www.g95.org/g95_status.html) , which include both large  
> production codes and codes from textbooks. It also compiles my codes.  
> There are a few classes on Fortran 90 or 95 using g95 as the main  
> compiler, so some professors think g95 is mature enough for new  
> programmers.

What about the huge list of bugs on the front page (which includes some really trivial ones!)? Even the compiler is written in an inappropriate language. <sigh>

>> No, I mean problems with your numerical code which have nothing to do  
> with  
>> maths. My main concern here is non-determinism from:  
>>  
>> 1. Out of bounds  
>> 2. Uninitialised variables  
>> 3. Aliasing  
>> ...  
>  
> The Fortran standard generally does not specify the required behavior  
> for such errors, but some compilers, including g95, do have good  
> error-checking, with the proper options enabled.

Yes, these days programming languages force you to initialise everything properly and enforce the checks, rather than requiring you to explicitly ask for them. This makes modern languages much easier to learn and use. Hence I recommend them.

>> As this is one students seem to have the most problems with. Most  
> compiler  
>> bugs probably also fall into this category. Memory leaks are  
> obviously

## Re: A question on Newton's Method

>> another important class of hassle which most numerical programmers  
>> shouldn't have to deal with.  
>  
> Using the ALLOCATABLE arrays of Fortran, one should not encounter  
> memory leaks.

I am not familiar with this. Looking at the Fortran source code from the book you were advertising, it contains explicit deallocations. Presumably omitting these would cause a memory leak?

>> An interactive programming environment is also particularly important  
> when  
>> starting out. Mathematica has an awesome graphical environment (the  
>> "notebook" front end) and OCaml has it's top-level.  
>  
> It may be helpful, but it is not essential. The edit-compile-debug  
> cycle does not bother me. Somehow people learned how to program, using  
> Fortran, even when they had to use punch cards and wait hours for their  
> results to come back.

I am trying to help the OP here, not ruin his life. :-)

>> Finally, the ability to visualise your results is hugely beneficial.  
> This is  
>> obviously easiest with Mathematica, followed by OCaml (the built-in  
>> Graphics module, or the lablGL OpenGL bindings) and then Fortran.  
>  
> There are many Fortran-callable graphics libraries, such as DISLIN.

That doesn't look like it'll take advantage of any modern graphics capabilities (e.g. hardware acceleration) which makes it unsuitable for most visualisation, except graph plotting.

This is a question of tractability again though. How many lines of Fortran 95 does it take to get the equivalent of these Mathematica "programs":

1000!

N[Pi, 100] (\* 100 digits of pi. \*)

Prime[1000] (\* The 1000th prime. \*)

Plot[Sin[x^2], {x, -5, 5}]

and do you really expect a new programmer to bother writing those hundreds of lines of Fortran just to test out some ideas?

Seriously, if the OP wants to study numerical robustness then the ability to use arbitrary-precision arithmetic easily is very useful. If the OP wants to get a better understanding of the Newton-Raphson method then a graphical representation will be of great benefit.

## Re: A question on Newton's Method

- > In general, your knowledge of Fortran and Fortran compilers appears to
- > be quite out of date, but that does not prevent you from making
- > sweeping generalizations about the language.

This is well known stuff, not personal belief. Fortran is still based on an archaic theoretical foundation, other languages aren't. Unless you are forced to use old code, I see no reason to use Fortran. I certainly see no reason for a new programmer to bother learning it. Does Fortran 95 have a single feature worth learning that isn't implemented in modern languages?

Also, we haven't touched on the lack of support for Fortran 95. Doing a Google search for "fortran 95" gives 85k pages compared to 508k for ocaml and 79M for Java. Searching for "fortran 2003" gives <4k pages. LOL.

—  
Dr Jon D Harrop, Flying Frog Consultancy  
<http://www.ffconsultancy.com>

---

### • *Follow-Ups:*

- ◆ ***Re: A question on Newton's Method***  
◇ *From:* kia
- ◆ ***Re: A question on Newton's Method***  
◇ *From:* beliavsky

### • *References:*

- ◆ ***A question on Newton's Method***  
◇ *From:* David M
- ◆ ***Re: A question on Newton's Method***  
◇ *From:* Jon Harrop
- ◆ ***Re: A question on Newton's Method***  
◇ *From:* beliavsky
- ◆ ***Re: A question on Newton's Method***  
◇ *From:* Jon Harrop
- ◆ ***Re: A question on Newton's Method***  
◇ *From:* beliavsky

- Prev by Date: ***Re: A question on Newton's Method***
- Next by Date: ***Re: A question on Newton's Method***
- Previous by thread: ***Re: A question on Newton's Method***
- Next by thread: ***Re: A question on Newton's Method***
- Index(es):
  - ◆ ***Date***
  - ◆ ***Thread***