Subject: Re: Looking for more ideas on code ...
Posted by Craig Markwardt on Tue, 01 Oct 2002 00:55:25 GMT
View Forum Message <> Reply to Message

jeyadev@wrc.xerox.bounceback.com (Surendar Jeyadev) writes:

```
>  I have a question about how best (style and function, if possible!) to
>  write code for a function that has limits that have to be treated in a
>  special way. Consider the function
>
>     f(x) = sin(x)/x
>
>  as an example. Now, if x is always a scalar, then on just tests to see
>  if it is zero, and then handle that special case using a if .. then ..
>  else construct. But, what if x can also be scalar? I have the following
>  code that works:
>
>  -----------------------------------------
>
>  function sinc, y
>
>
>  if(n_elements(y) eq 1) then begin                 ; y is a scalar
>    if(y eq 0.0) then profile = 1.0 else begin
>      profile = sin(y)/y
>    endelse
>   endif else begin                              ; y is a vector
>    zeros = where(y eq 0.0, ind)
>    if(ind gt 0) then y(zeros) = 1.0e-10       ; set zeroes to a small quantity
>    profile = sin(y)/y
>  endelse
>
>  profile = profile*profile/a0
>
>
>  return, profile
>
>  end
>
>  -----------------------------------------
>
>  I guess the one can always set
>
>    profile(zeros) = 1.0
>
>  to handle the more general cases. But, the real question is there
>  a better way than
>
```

```
>     zeros = where(y eq 0.0, ind)
>     if(ind gt 0) then y(zeros) = "special values"
>     notzeros = where(y ne 0.0, ind)
>     if(ind gt 0) then y(notzeros) = "general definition"
>
>  I do understand that one should not compare reals, etc., but I will
>  clean up the numerics later.
```

First of all, this is a perfectly good time to compare reals. The discontinuity only exists at zero, no where else.

Second of all, you can simplify your logic a little, by pre-filling the array with the "special case:"

```
  profile = y*0 + 1.   ;; Tricky way to get array filled with zeroes
  wh = where(y NE 0, ct)
  profile(wh) = sin(y(wh))/y(wh)
```

That's not too bad. You can get trickier too, and in fact cheat with this little doozy, which only works for an even function:

```
  sz = size(y) & isdouble = sz(sz(0)+1) EQ 5
  ymin = (machar(double=isdouble)).xmin
  yp = abs(y)+ymin

  profile = sin(yp)/yp
```

This will get at maximum only 2 ulps of error, and even then only if your values of Y are near 10^{-308}. It works by adding a small value to ABS(Y), so that the sinc evaluation is never exactly equal to zero. The top two lines are needed to determine correct value to add, float vs. double. This is a bit too sneaky.

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------