Subject: Re: how to get an array of variable length arrays
Posted by dave on Mon, 28 Nov 1994 23:22:15 GMT
View Forum Message <> Reply to Message

>>>> > "James" == James Tappin <sjt@xun8.sr.bham.ac.uk> writes:
In article <3bc8np$q64@sun4.bham.ac.uk> sjt@xun8.sr.bham.ac.uk (James Tappin) writes:

James> greec@essex.ac.uk wrote: : Does anyone know how you can
James> create an array of variable length : arrays?.

James> : e.g in C you could do it like this:

James> : float a[10],b[5],c[14]; : float *arrays[3];

James> : arrays[0]=a; arrays[1]=b; arrays[2]=c;

James> : Thanks in advance for any help you can give.

James> : Chris

James> : greec@essex.ac.uk

James> The only thing I can think of would be to use a structure:
James> e.g. for you example

James> a = fltarr(10) b = fltarr(5) c = fltarr(14) arrays = {f1:a,
James> f2:b, f3:c}

James> You can then access the fields numerically by using the
James> form: val = arrays.(n)(m)

James> But you can't then replace the f3 field with a 6 element
James> array, you would have to make a new structure.

The structure approach is the simplest. You could get more
flexibility with handles:

a = fltarr(10)& b=fltarr(5) &c=fltarr(14)
arrays=lonarr(3)

array(0)=handle_create(/value=a)
array(1)=handle_create(/value=b)
array(2)=handle_create(/value=c)

Of course, the handle mechanism is not as simple or elegant as the C

pointer mechanism, and you need to use handle_value to access the elements.:

; get j=array(x,*):

handle_value,array(x), j

--

David Fenyes dave@image6.med.uth.tmc.edu
University of Texas Medical School Houston, Texas