Subject: Re: zero-padding an array of arbitrary dimensionality (replacing execute in vm)
Posted by Allan Whiteford on Thu, 19 Jul 2007 16:37:26 GMT
View Forum Message <> Reply to Message

Vince,

data=fltarr(43,45,23,12) ; <-- arbitrary
data[1,1,1,1]=1000. ; <-- so we know we get the right answer

tmp=size(data)
tmp=tmp[1:tmp[0]]
idx=1
for i=0,n_elements(tmp)-2 do idx=idx+product(tmp[0:i])

print,data[idx] ; We get element [1,1,1,1]

Helpful?

Probably doesn't work for 1D arrays.

Thanks,

Allan

hradilv wrote:
> I would like to zero-pad an array programmatically without knowing in
> advance what the dimensionality is of the array.
>
> For example, in 2D, for data = some fltarr (31,31) I could do
> dims = size(data,/dimensions)
> zpad = fltarr(dims[0]+1,dims[1]+1)
> zpad[1,1] = data
>
> For arb. dimensionality I have:
>
> dsize = size(data)
> ndim = dsize[0]
> dim = dsize[1:ndim]
> dtmp = make_array(dim+2,value=0,type=dsize[ndim+1])
>
> cmd = 'dtmp['
> for n=0L, ndim-1 do cmd = cmd+'1,'
> cmd = strmid(cmd,0,strlen(cmd)-1)+']=data'
> result = execute(cmd)
>
> But this won't work in the vm. So I need to somehow figure out the
> position of the [1,1,1,...] index for an arbitrary dimensionality.

>
> Clear enough? TIA!
> Vince
>