Subject: Array indexing: what is IDL doing?
Posted by Conor on Fri, 06 Jun 2008 16:50:33 GMT
View Forum Message <> Reply to Message

So I want to pull out a small subsection from within a larger idl
array. I want to do something like this:

```
bigarr = findgen(500,100000)
bigx = [indgen(50)+100,indgen(25)+200]
bigy = indgen(10000)+50000

res = bigarr[bigx,bigy]
```

However IDL doesn't like that and says:

```
% All array subscripts must be same size. Var = BIGARR
% Execution halted at: $MAIN$
```

So I have to come up with another way to do it. My two thoughts were
the most straight forward way:

```
tarr = bigarr[bigx,*]
res = tarr[*,bigy]
```

and my other thought was to pre-build the indexes I want to extract,
and so I came up with this relatively simple command:

```
cx = n_elements(wx)
cy = n_elements(wy)
nx = n_elements(bigarr[*,0])
ind = transpose(rebin(wy,cy,cx))*nx + rebin(wx,cx,cy)
res = bigarr[ind]
```

I then wrote a little script to compare execution times, and I find
that if you are only extracting a small subsection (say, 500x10,000)
the index building method takes .04 seconds, compared to .11 seconds
for indexing bigarr twice. I'm really curious though why there's a
difference. Does it have something to do with using the asterix in
indexing? Is it just because it has two make two copies when I use
the first method? Since I'm extracting the same region either way,
I'd expect them to take about the same amount of time...