Subject: where help

Posted by Phil Williams on Mon, 27 Jan 1997 08:00:00 GMT

View Forum Message <> Reply to Message

How do I get the x,y coords from the result of where?

Here's what I did:

IDL> t = where(slice eq max(slice))

IDL> print, slice(t)

28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28	28	28	28	28	28	28
28	28	28						

IDL> print,size(slice)

IDL> y = t/255

IDL > x = t - (y*255)

IDL> print,slice(x,y)

0	0	0	0	0	0	10	10	10
11	9	10	0	10	10	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0						

I also tried

IDL> print,slice(y,x)

U	Ü	Ü	Ü	Ü	Ü	Ü	0	Ü
0	0	0	0	12	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	10	11	19
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	10	0	0
0	10	0						

IDL> print,max(slice)

28

What is going on??? Any help would be GREATLY appreciated!

Subject: Re: where help Posted by davidf on Mon, 27 Jan 1997 08:00:00 GMT View Forum Message <> Reply to Message

Phil Williams <williams@irc.chmcc.org> writes:

> How do I get the x,y coords from the result of where?

>

- > Here's what I did:
- > IDL> t = where(slice eq max(slice))

I'm really glad you asked this question! I am sure there is some nifty algorithm that everyone learns in CS101. But since I taught myself how to program, I don't know it. And I've always wanted to figure it out. So here goes...

Here is what I know (for a 2D array!). Arrays are stored in row order, so the column index varies the fastest. Suppose I create a 3 column by 4 row array and get its size, like this:

```
data = FINDGEN(12)
data = REFORM(data, 3, 4)
PRINT, data
 0.00000
            1.00000
                       2.00000
 3.00000
            4.00000
                       5.00000
 6.00000
            7.00000
                       8.00000
 9.00000
            10.0000
                       11.0000
s = SIZE(data)
```

OK, suppose I now do this:

index = WHERE(data EQ 6)

The WHERE function returns the 1D index into the 2D array.

```
PRINT, index 6
```

So, the number 6 is located in index 6 (the 7th number in the array with zero-based subscripting). What is its 2D subscript? Well, if I divide index by how many columns there are in the array, and then take the whole part of that number, I will know its row number. In IDL terms:

```
row = FIX(FLOAT(index)/s(1))
PRINT, row
2
```

To find the column number, I multiply the row number times the number of columns in the array, and subtract that value from the index. Again, in IDL terms:

```
col = index - (row * s(1))
PRINT, col
0
```

That seem's right. Let's test it in the real world.

```
seed = -1L
data = RANDOMU(seed, 50, 100)
s = SIZE(data)
index = 124
PRINT, data(index)
0.973124
```

Now for the test:

```
row = FIX(FLOAT(index)/s(1))
col = index - (row * s(1))
PRINT, col, row
24 2
PRINT, data(col, row)
0.973124
```

Hooray!

I don't know if its the algorithm they teach in CS101, but it works for me. :-)

Cheers!

David

David Fanning, Ph.D.

Fanning Software Consulting

2642 Bradbury Court, Fort Collins, CO 80521 Phone: 970-221-0438 Fax: 970-221-4762

E-Mail: davidf@dfanning.com

Coyote's Guide to IDL Programming: http://www.dfanning.com
