Subject: READ_ASCII - accessing data from structures
Posted by m.doyle on Wed, 12 May 2004 17:14:05 GMT
View Forum Message <> Reply to Message

Hello all,

I wonder if anyone might be able to help me with this?

I'm using the IDL READ_ASCII function to read in a semicolon seperated file, after which I'd like to access individual elements of that data file. From the RSI website, I see that I can access the data fields (i.e. each column of data) by using, for example;

print, mydata.(4)

using the Variable_Name.(Tag_Index) method.

Could someone tell me how to get to the individual elements of mydata.(4)?? Using the above I can only get a large stream of numbers, I need to use them one by one.

Many thanks for your help!

Martin

Subject: Re: read_ascii
Posted by David Fanning on Wed, 08 Dec 2004 20:54:19 GMT
View Forum Message <> Reply to Message

Francois writes:

> I use the following code to read an ascii file:
>> filename = dialog_pickfile(path='c:\')
>> data = READ_ASCII(filename, DATA_START=1, DELIMITER= ',')
>> data = data.field1
>
> I want data to become an array instead of a structure.
>
> The problem is that sometimes the array is contained in field01,
> sometimes in field01.
> How come it varies ?

I'm not sure I see the variation in the question, but in any case I don't know why it varies. If it *is* varying, why not try this:

data = READ_ASCII(filename, DATA_START=1, DELIMITER= ',')

data = data.(0)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: read_ascii
Posted by Rick Towler on Wed, 08 Dec 2004 21:21:22 GMT
View Forum Message <> Reply to Message

> The problem is that sometimes the array is contained in field01,
> sometimes in field01.

> How come it varies ?

You mean "field1" and "field01".

When confronted with less than 10 fields, READ_ASCII will return field1-field9, when you have more than 10, field01-field99.

It is unfortunate that READ_ASCII behaves this way but David's suggestion of addressing the structure by tag index instead of tag name will solve your problem.

-Rick

Subject: Re: read_ascii
Posted by Benjamin Tupper on Thu, 09 Dec 2004 13:52:12 GMT
View Forum Message <> Reply to Message

Rick Towler wrote:

>
>> The problem is that sometimes the array is contained in field01,
>> sometimes in field01.
>
>
>> How come it varies ?
>
>

> You mean "field1" and "field01".
>
> When confronted with less than 10 fields, READ_ASCII will return
> field1-field9, when you have more than 10, field01-field99.
>
> It is unfortunate that READ_ASCII behaves this way but David's
> suggestion of addressing the structure by tag index instead of tag name
> will solve your problem.
>

Hi,

You could define the names of the fields using ASCII_TEMPLATE before calling READ_ASCII. You can define the template without using ASCII_TEMPLATE - a little study of the structure returned by ASCII_TEMPLATE should help.

Alternatively, you could modify READ_ASCII (call it something else, though, like MY_READ_ASCII). You can force the routine to always use N digits in the field names by modifying the following line (line 874 in my version) ....

```
digits_str = $
 strtrim(string(strlen(strtrim(string(fieldCountUse),2))),2)
```

to something like this...

```
my_dig_len = fieldCountUse > 2
digits_str = $
 strtrim(string(strlen(strtrim(string(my_dig_len),2))),2)
```

On a related note, I have drifted away from treating columnated ASCII data files as nColumns of vectors. Instead, I treat them as vectors of structures where each row is a record and each column is a field of the record. This works fine for flatly organized data and I find a vector of structures MUCH easier to manage in IDL than a structure of vectors. If you are interested seeing my version of READ_ASCII then shoot me an email.

Cheers,
Ben