Subject: Re: basic array-structure understanding question
Posted by julia.walterspiel on Wed, 19 Nov 2008 15:49:16 GMT
View Forum Message <> Reply to Message

On 19 Nov., 16:41, "m_schell...@hotmail.com" <mschell...@gmail.com>
wrote:
> I am sure David has an appropiate link for this issue,
> but for now:
>
> On 19 Nov., 15:21, julia.waltersp...@gmail.com wrote:
>
>
>
>
>
>> hi everybody
>
>> This is what does not get into my head:
>
>> I have one array containing my data.
>> I have a second array containing the times when the data was
>> collected.
>> They have the same lenght.
>
>> I put them into one structure because we all like structures and I
>> learned about their advantages.
>
>> now:
>
>> IN SHORT are those arrays linked somehow? in other words: when working
>> with a structure, does IDL know which value matches the corresponding
>> date, given the data and the date array have the same length. Are
>> those arrays somehow connected or are they completely independent from
>> each other?
>
> As you described, they are semantically linked.
> I. e. structure.data[i] corresponds to structure.time[i] for all i.
> For IDL there is no such association.
>
>> IN LONG: if I want to get e.g. all the data from all Januarys from
>> 2000 to 2007, can I do it somehow like
>
>> all_jan = structure.data (where(structure.time EQ 200?01??????))..
>> then it would automatically "select" only the right values. (and then
>> I would have to make a separate array of the corresponding dates, make
>> those two new arrays a structure so that I can plot data vs. date all
>> in one?)
>

>> or do I have to do it via some indexing, like:
>> all_jan = where (structure.time EQ 200?01??????))
>
>> and then apply the index to my array of data?
>
> Both ways are fine. In the first case the index array (output of
> WHERE) is a temporary variable,
> in the second case you make it a (normal) variable named all_jan.
>
> In the first case your selected data is in all_jan.
> In the second case the indices are in all_jan and you have to create
> the selected data by using
> structure.data[all_jan]
>
> It is quite simple, no magic.
> What might seem like magic is the
> EQ (or any other logical) operator working on arrays.
> WHERE just returns the inidices where the elements of the
> expression (structure.time EQ 200?01??????) (which is a byte array)
> are 'true'
> (depends on the data type. For integers including byte 'true' is NE
> 0).
> (you are aware that 'structure.time EQ 200?01??????' is pseudo code)
>
> To understand this better, do:
>
> IDL> eqArr = structure.time EQ 200?01??????
> IDL> whereArr = where(eqArr)
> IDL> print,eqArr
> IDL> print,whereArr
>
> You should however use the second case and check the result for -1
> (IF all_jan[0] EQ -1 THEN ...)
> as this is the result of WHERE if no elements are 'true'.
>
> HTH,
> Marc- Zitierten Text ausblenden -
>
> - Zitierten Text anzeigen -

thanks Marc! the knot is coming undone..
it's a pity that IDL does not associate the two arrays. But I guess
nobody's perfect :) (or is there a way to TELL IDL to associate them?)
cheers for the explanation. I'll go hunt around for some more of
David's articles on this topic.
juls