Subject: Function with something like [2:*] as input
Posted by andry on Thu, 09 Jan 2014 20:06:35 GMT
View Forum Message <> Reply to Message

Hi,

Could anybody help on how I can write a function so that the input will be something like the indices of an array:

For example, my function would accept a call of this format:

a= my_function([1:3]) ; to say that it will process element 1 to 3 of an array
a= my_function([[1:*]) ; to say that it will process everything from element 1 to the end. This is because most of the time, I would not know how many data I have in some array.

My problem is that I can not see how I can read that input from within the function.

I am not sure this makes sense but it will facilitate the analysis of my data.

Thanks in advance for your help,

Andry

Subject: Re: Function with something like [2:*] as input
Posted by Matthew Argall on Thu, 09 Jan 2014 20:57:31 GMT
View Forum Message <> Reply to Message

Here is one way to do it. Create the function below, then

```
IDL> array = findgen(20)
IDL> print, SubArray(array, 1, 3)
IDL> print, SubArray(array, 1, N_Elements(array)-1)
IDL> print, SubArray(array, 1, -1)


function SubArray, array, index1, index2
  return, array[index1:index2]
end
```

Subject: Re: Function with something like [2:*] as input
Posted by andry on Fri, 10 Jan 2014 12:34:11 GMT
View Forum Message <> Reply to Message

On Thursday, January 9, 2014 5:27:31 PM UTC-3:30, Matthew Argall wrote:
> Here is one way to do it. Create the function below, then

>
>
>
> IDL> array = findgen(20)
>
> IDL> print, SubArray(array, 1, 3)
>
> IDL> print, SubArray(array, 1, N_Elements(array)-1)
>
> IDL> print, SubArray(array, 1, -1)
>
>
>
>
>
> function SubArray, array, index1, index2
>
> return, array[index1:index2]
>
> end

Thanks,

I did not know about the "-1" index.

Andry

Subject: Re: Function with something like [2:*] as input
Posted by chris_torrence@NOSPAM on Fri, 10 Jan 2014 15:28:11 GMT
View Forum Message <> Reply to Message

Hi Andry,

If you really wanted to get fancy, and you have IDL 8.0 or later, you could use operator overloading. You could create an object class that is a subclass of IDL_Object, and then implement the _overloadBracketsRightSide method. Then you could do things like:

myfunc = MyFunctionClass()
a = myfunc[1:3] ; looks just like array indexing!

See the examples at:
http://www.exelisvis.com/docs/Overloading_the_Array_In.html

Cheers,
Chris
ExelisVIS

Subject: Re: Function with something like [2:*] as input
Posted by David Fanning on Fri, 10 Jan 2014 15:39:18 GMT
View Forum Message <> Reply to Message

Chris Torrence writes:

> If you really wanted to get fancy, and you have IDL 8.0 or later, you could use operator overloading. You could create an object class that is a subclass of IDL_Object, and then implement the _overloadBracketsRightSide method. Then you could do things like:
>
> myfunc = MyFunctionClass()
> a = myfunc[1:3] ; looks just like array indexing!
>
> See the examples at:
> http://www.exelisvis.com/docs/Overloading_the_Array_In.html

Right. I was going to mention that just after I explained the value of using positional and keyword parameters. ;-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")