Subject: Re: Another strsplit question
Posted by MKatz843 on Tue, 20 Jan 2004 20:49:08 GMT
View Forum Message <> Reply to Message

kneller@nist.gov (Larry Kneller) wrote in message
news:<96e91d8d.0401200800.34fa60fa@posting.google.com>...
> Most functions in IDL will operate on arrays.
> Is there a way to make strsplit allow this?
>
> e.g.
> s = ['I see the computer.','I am typing a message.',$
> 'This is the idl-pvwave newsgroup.']
>
> segs = strsplit(s,/extract,/some_option)
>
> print,segs[0,0],segs[1,0],segs[2,0],segs[2,4]
>
> RESULT WOULD BE:
>
> I I This newsgroup.

The difficulty here is that since every input string is different,
every row of the output array will want to have a different number of
values. So it's ambiguous how you'd write it. Would you look for the
row with the most elements and make all rows be that long? That's one
way. Or would you have the output be a pointer array where each
element points to an array of strings. That's probably more memory
efficient in the long run, but some people might not like dealing with
pointers, and you'd run the risk of asking for array elements that
don't exist.

Looks like you need to write your own function here to fit your needs.
That's pretty straightforward to do in a case like this.

M. Katz

Subject: Re: Another strsplit question
Posted by kneller on Wed, 21 Jan 2004 16:31:16 GMT
View Forum Message <> Reply to Message

MKatz843@onebox.com (M. Katz) wrote in message
news:<4a097d6a.0401201249.3299ca9b@posting.google.com>...
> kneller@nist.gov (Larry Kneller) wrote in message
news:<96e91d8d.0401200800.34fa60fa@posting.google.com>...
>> Most functions in IDL will operate on arrays.
>> Is there a way to make strsplit allow this?
>>

>> e.g.
>> s = ['I see the computer.','I am typing a message.',$
>> 'This is the idl-pvwave newsgroup.']
>>
>> segs = strsplit(s,/extract,/some_option)
>>
>> print,segs[0,0],segs[1,0],segs[2,0],segs[2,4]
>>
>> RESULT WOULD BE:
>>
>> I I This newsgroup.
>
> The difficulty here is that since every input string is different,
> every row of the output array will want to have a different number of
> values. So it's ambiguous how you'd write it. Would you look for the
> row with the most elements and make all rows be that long? That's one
> way. Or would you have the output be a pointer array where each
> element points to an array of strings. That's probably more memory
> efficient in the long run, but some people might not like dealing with
> pointers, and you'd run the risk of asking for array elements that
> don't exist.
>
> Looks like you need to write your own function here to fit your needs.
> That's pretty straightforward to do in a case like this.
>
> M. Katz

Thanks for the suggestions! I think the array of pointers makes more
sense for my personal use. That would easily allow a ragged array
of 'strings'.
