Subject: Re: allocating data to different array on each loop step
Posted by R.Bauer on Sat, 12 Nov 2005 11:08:43 GMT
View Forum Message <> Reply to Message

snfinder@naver.com wrote:

> Dear all,
>
> I just want to allocate data(computed in each loop step) to different
> array in each step.
>

if n_elements(arr) eq 0 then arr=value else arr=[arr,value]

later on you probably have to do somewhere result=temporary(arr)

cheers
Reimar

> for example,
> while ...
>
> ..
> ..
> data=...
>
> a0=data
> .
> .
> endwhile
>
>
> in other words
> loop step 1: a1=data
> loop step 2: a2=data
> loop step 3: a3=data
> .
> .
> .
> end of loop
>
>
> I need new array on each step but I can not know the size of array(data
> computed) in advace.
> It would be differnt on each step.
>

>
> To sum up,
> I need the generation of array(different name ex) a0,a1,a2....on each
> loop step.
>
>
> Thank you.
>
> Help me~ You can give a bright to a blind(IDL blind like me ^^;;).
> SOS~~

Subject: Re: allocating data to different array on each loop step
Posted by snfinder@naver.com on Sat, 12 Nov 2005 12:10:25 GMT
View Forum Message <> Reply to Message

Thank you for replying

But I can't understand what you wrote.
Tha data in loop is also array not a single value
Therefore I will have several arrays after loop operation.

What I exactly want is to save the array(data computed in each step) to
different array ecah loop step.

loop 1
 array1=data computed

loop 2
 array2=data computed
.
.
.
.

loop 24(end of loop forexample)
 array24=data computed

So, finally I have 24 different arrays allocated different data.

Thank you
Can you explain more details again?

Subject: Re: allocating data to different array on each loop step

Posted by David Fanning on Sat, 12 Nov 2005 16:25:13 GMT
View Forum Message <> Reply to Message

snfinder@naver.com writes:

> Tha data in loop is also array not a single value
> Therefore I will have several arrays after loop operation.

I would do something like this:

```
arrays = PtrArr(24)
FOR j=0,23 DO BEGIN
  ...calculation to obtain array of whatever size
  arrays[j] = Ptr_New(thisLoopArray, /No_Copy)
ENDFOR
```

Now you have a pointer array. Each element in the array
is a pointer to another array of variable length.

```
   array1 = *arrays[0]
   array2 = *arrays[1]
   ...
   array24 = *arrays[23]
```

Be sure to free your pointers when you are done with them:

```
   Ptr_Free, arrays
```

Cheers,

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

Subject: Re: allocating data to different array on each loop step
Posted by snfinder@naver.com on Mon, 14 Nov 2005 01:50:33 GMT
View Forum Message <> Reply to Message

Thank you Coyote~ ^^

But If the number of loop step is so large,

```
   array1 = *arrays[0]
   array2 = *arrays[1]
   ...
   array24 = *arrays[23]
```

this is very unefficient.

I want to excute this automatically.

And I can't know the number of loop step before end of all steps.
um...ok can find out it if I excute loop twice.....ok..

Thank you ~

Subject: Re: allocating data to different array on each loop step
Posted by Richard French on Mon, 14 Nov 2005 02:02:34 GMT
View Forum Message <> Reply to Message

On 11/13/05 8:50 PM, in article
1131933033.894813.142800@z14g2000cwz.googlegroups.com, "snfinder@naver.com"
<snfinder@naver.com> wrote:

> Thank you Coyote~ ^^
>
> But If the number of loop step is so large,
>   array1 = *arrays[0]
>   array2 = *arrays[1]
>   ...
>   array24 = *arrays[23]
> this is very unefficient.
>
> I want to excute this automatically.
>
> And I can't know the number of loop step before end of all steps.
> um...ok can find out it if I excute loop twice.....ok..
>
> Thank you ~
>

Well, you can use the EXECUTE command if you want to define the variables on
the fly:

```
pro arrays
for n=1L,10 do begin
;create arrays of different sizes
      array=findgen(n)#findgen(10*n)
; construct a command to execute, defining the new variable name
      cmd=string(n,format='("array",I0,"=array")')
      print,cmd
      result=execute(cmd)
```

```
endfor
help
end


IDL> arrays
array1=array
array2=array
array3=array
array4=array
array5=array
array6=array
array7=array
array8=array
array9=array
array10=array
% At ARRAYS             8 /Users/rfrench/arrays.pro
%    $MAIN$
ARRAY          FLOAT     = Array[10, 100]
ARRAY1         FLOAT     = Array[1, 10]
ARRAY10        FLOAT     = Array[10, 100]
ARRAY2         FLOAT     = Array[2, 20]
ARRAY3         FLOAT     = Array[3, 30]
ARRAY4         FLOAT     = Array[4, 40]
ARRAY5         FLOAT     = Array[5, 50]
ARRAY6         FLOAT     = Array[6, 60]
ARRAY7         FLOAT     = Array[7, 70]
ARRAY8         FLOAT     = Array[8, 80]
ARRAY9         FLOAT     = Array[9, 90]
CMD            STRING    = 'array10=array'
N              LONG      =          11
RESULT         INT       =       1
```

Dick French

Subject: Re: allocating data to different array on each loop step
Posted by snfinder@naver.com on Mon, 14 Nov 2005 03:25:52 GMT
View Forum Message <> Reply to Message

A thousand thanks~~ !!
Thank you Thank you very much ^^! Richard G. French~

Ha Ha Ha~~
It makes me really happy~~~

Subject: Re: allocating data to different array on each loop step
Posted by Richard French on Mon, 14 Nov 2005 03:31:08 GMT
View Forum Message <> Reply to Message

On 11/13/05 10:25 PM, in article
1131938752.543314.79260@g47g2000cwa.googlegroups.com, "snfinder@naver.com"
<snfinder@naver.com> wrote:

> A thousand thanks~~ !!
> Thank you Thank you very much ^^! Richard G. French~
>
> Ha Ha Ha~~
> It makes me really happy~~~

>

Well, that is a cheerful message to get just before the start of a long
week! Thanks to you!
DF

Subject: Re: allocating data to different array on each loop step
Posted by David Fanning on Mon, 14 Nov 2005 04:52:56 GMT
View Forum Message <> Reply to Message

Richard G. French writes:

> Well, you can use the EXECUTE command if you want to define the variables on
> the fly:
>
>
>
> pro arrays
> for n=1L,10 do begin
> ;create arrays of different sizes
> array=findgen(n)#findgen(10*n)
> ; construct a command to execute, defining the new variable name
> cmd=string(n,format='("array",I0,"=array")')
> print,cmd
> result=execute(cmd)
> endfor
> help
> end
>
>
> IDL> arrays
> array1=array

> array2=array
> array3=array
> array4=array
> array5=array
> array6=array
> array7=array
> array8=array
> array9=array
> array10=array
> % At ARRAYS 8 /Users/rfrench/arrays.pro
> % $MAIN$
> ARRAY FLOAT = Array[10, 100]
> ARRAY1 FLOAT = Array[1, 10]
> ARRAY10 FLOAT = Array[10, 100]
> ARRAY2 FLOAT = Array[2, 20]
> ARRAY3 FLOAT = Array[3, 30]
> ARRAY4 FLOAT = Array[4, 40]
> ARRAY5 FLOAT = Array[5, 50]
> ARRAY6 FLOAT = Array[6, 60]
> ARRAY7 FLOAT = Array[7, 70]
> ARRAY8 FLOAT = Array[8, 80]
> ARRAY9 FLOAT = Array[9, 90]
> CMD STRING = 'array10=array'
> N LONG = 11
> RESULT INT = 1

Humm. I thought he was looking for "efficient". Oh, well,
I guess the old garage sale adage holds for science
concepts, too: What's junk to me, is a treasure for you! :-)

Cheers,

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

Subject: Re: allocating data to different array on each loop step
Posted by Paolo Grigis on Mon, 14 Nov 2005 08:49:21 GMT
View Forum Message <> Reply to Message

snfinder@naver.com wrote:
> Thank you Coyote~ ^^
>
> But If the number of loop step is so large,
> array1 = *arrays[0]

>   array2 = *arrays[1]
>   ...
>   array24 = *arrays[23]
> this is very unefficient.
Do you really want to have 24 differently named arrays?
You don't need named arrays to access the data stored
in your pointers to arrays, it's enough to write
something like

(*arrays[k])[i]

Ciao,
Paolo

>
> I want to excute this automatically.
>
> And I can't know the number of loop step before end of all steps.
> um...ok can find out it if I excute loop twice.....ok..
>
> Thank you ~
>
