
Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [Brian Daniel](#) on Thu, 29 Sep 2011 11:50:25 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 29, 5:10 am, voidspace <jhkim...@gmail.com> wrote:

> Hello folks,

>

> I have a question at beginner's level. When I saved multiple variables

> with same head but different tails in IDL, I found it is tedious to

> type all of them in as follows.

>

> SAVE, sst_cccma_cgcm3_1, sst_cnrm_cm3, sst_csiro_mk3_0,

> sst_gfdl_cm2_0, sst_gfdl_cm2_1, \$

> sst_iap_fgoals1_0_g, sst_inmcm3_0, sst_ipsl_cm4,

> sst_miroc3_2_medres, sst_mpi_echam5, \$

> sst_mri_cgcm2_3_2a, sst_ncar_ccsm3_0, sst_ncar_pcm1,

> sst_ukmo_hadcm3, sst_ukmo_hadgem1, \$

> FILENAME='sst_cmip3_20c3m.sav'

>

> The '/All' keyword may help, but there are more variables that I don't

> want to save in. This aroused my curiosity, so I post my question here

> to know whether there is a workaround.

>

> In fact, I tried to find an advanced way by myself, but found it's

> difficult to know without an expert's guidance.

>

> An attempt with my best knowledge is as follows. As all model names

> from 'cccma_cgcm3_1' to 'ukmo_hadgem1' are saved in a string array

> 'model', I considered the 'EXECUTE' command.

>

> One prior step I did was saving 15 variable names in one single string

> array 'sst'.

>

> IDL> sst=STRARR(N_elements(model))

> IDL> FOR i=0, N_elements(model)-1 DO sst[i]='sst_'+model[i]

>

> Now, 'sst' contains 15 different variable names that I want to save

> in. Then, tried the following

>

> IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', "+sst[i])

>

> but in stuck because I have no idea how to make an implicit loop for

> 'sst[i]'.

>

> My attempt seems not a right way. I also thought it would be

> convenient if there is implicit do-loop like (sst(i),i=1,15) in

> Fortran, but I immediately got that it's nothing but a stupid idea.

>
> I thought over and googled as well but all were in vain.
>
> Can anybody suggest me a nice way to simplify the tedious command
> aforementioned?
>
> Best,
> John

You've already done the hard part, which is the string manipulation to create your 'sst' array. The final step is to use StrJoin to bring the array into one string joined by ', '. See below (UNTESTED).

```
result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,'  
,/Single) )
```

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [d.poreh](#) on Thu, 29 Sep 2011 12:03:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 29, 4:50 am, "Brian J. Daniel" <Daniels...@yahoo.com> wrote:

> On Sep 29, 5:10 am, voidspace <jhkim...@gmail.com> wrote:

>
>
>
>
>
>
>
>
>
>
>
>> Hello folks,
>
>> I have a question at beginner's level. When I saved multiple variables
>> with same head but different tails in IDL, I found it is tedious to
>> type all of them in as follows.
>
>> SAVE, sst_cccma_cgcm3_1, sst_cnrm_cm3, sst_csiro_mk3_0,
>> sst_gfdl_cm2_0, sst_gfdl_cm2_1, \$
>> sst_iap_fgoals1_0_g, sst_inmcm3_0, sst_ipsl_cm4,
>> sst_miroc3_2_medres, sst_mpi_echam5, \$
>> sst_mri_cgcm2_3_2a, sst_ncar_ccsm3_0, sst_ncar_pcm1,
>> sst_ukmo_hadcm3, sst_ukmo_hadgem1, \$
>> FILENAME='sst_cmip3_20c3m.sav'
>
>> The '/All' keyword may help, but there are more variables that I don't

```

>> want to save in. This aroused my curiosity, so I post my question here
>> to know whether there is a workaround.
>
>> In fact, I tried to find an advanced way by myself, but found it's
>> difficult to know without an expert's guidance.
>
>> An attempt with my best knowledge is as follows. As all model names
>> from 'cccma_cgcm3_1' to 'ukmo_hadgem1' are saved in a string array
>> 'model', I considered the 'EXECUTE' command.
>
>> One prior step I did was saving 15 variable names in one single string
>> array 'sst'.
>
>> IDL> sst=STRARR(N_elements(model))
>> IDL> FOR i=0, N_elements(model)-1 DO sst[i]='sst_'+model[i]
>
>> Now, 'sst' contains 15 different variable names that I want to save
>> in. Then, tried the following
>
>> IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', "+sst[i] )
>
>> but in stuck because I have no idea how to make an implicit loop for
>> 'sst[i]'.
>
>> My attempt seems not a right way. I also thought it would be
>> convenient if there is implicit do-loop like (sst(i),i=1,15) in
>> Fortran, but I immediately got that it's nothing but a stupid idea.
>
>> I thought over and googled as well but all were in vain.
>
>> Can anybody suggest me a nice way to simplify the tedious command
>> aforementioned?
>
>> Best,
>> John
>
> You've already done the hard part, which is the string manipulation to
> create your 'sst' array. The final step is to use StrJoin to bring
> the array into one string joined by ', '. See below (UNTESTED).
>
> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,',
> ',/Single) )

```

Can you prepare a simple example? method seems quite good for me!
 Cheers,
 Dave

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [Brian Daniel](#) on Thu, 29 Sep 2011 12:45:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 29, 8:03 am, Dave Poreh <d.po...@gmail.com> wrote:

> On Sep 29, 4:50 am, "Brian J. Daniel" <Daniels...@yahoo.com> wrote:

>

>

>

>

>

>

>

>

>

>> On Sep 29, 5:10 am, voidspace <jhkim...@gmail.com> wrote:

>

>>> Hello folks,

>

>>> I have a question at beginner's level. When I saved multiple variables

>>> with same head but different tails in IDL, I found it is tedious to

>>> type all of them in as follows.

>

>>> SAVE, sst_cccma_cgcm3_1, sst_cnrm_cm3, sst_csiro_mk3_0,

>>> sst_gfdl_cm2_0, sst_gfdl_cm2_1, \$

>>> sst_iap_fgoals1_0_g, sst_inmcm3_0, sst_ipsl_cm4,

>>> sst_miroc3_2_medres, sst_mpi_echam5, \$

>>> sst_mri_cgcm2_3_2a, sst_ncar_ccsm3_0, sst_ncar_pcm1,

>>> sst_ukmo_hadcm3, sst_ukmo_hadgem1, \$

>>> FILENAME='sst_cmip3_20c3m.sav'

>

>>> The '/All' keyword may help, but there are more variables that I don't

>>> want to save in. This aroused my curiosity, so I post my question here

>>> to know whether there is a workaround.

>

>>> In fact, I tried to find an advanced way by myself, but found it's

>>> difficult to know without an expert's guidance.

>

>>> An attempt with my best knowledge is as follows. As all model names

>>> from 'cccma_cgcm3_1' to 'ukmo_hadgem1' are saved in a string array

>>> 'model', I considered the 'EXECUTE' command.

>

>>> One prior step I did was saving 15 variable names in one single string

>>> array 'sst'.

>

>>> IDL> sst=STRARR(N_elements(model))

>>> IDL> FOR i=0, N_elements(model)-1 DO sst[i]='sst_'+model[i]

>

```

>>> Now, 'sst' contains 15 different variable names that I want to save
>>> in. Then, tried the following
>
>>> IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', "+sst[i] )
>
>>> but in stuck because I have no idea how to make an implicit loop for
>>> 'sst[i]'.
>
>>> My attempt seems not a right way. I also thought it would be
>>> convenient if there is implicit do-loop like (sst(i),i=1,15) in
>>> Fortran, but I immediately got that it's nothing but a stupid idea.
>
>>> I thought over and googled as well but all were in vain.
>
>>> Can anybody suggest me a nice way to simplify the tedious command
>>> aforementioned?
>
>>> Best,
>>> John
>
>> You've already done the hard part, which is the string manipulation to
>> create your 'sst' array. The final step is to use StrJoin to bring
>> the array into one string joined by ', '. See below (UNTESTED).
>
>> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,',
>> ',/Single) )
>
> Can you prepare a simple example? method seems quite good for me!
> Cheers,
> Dave

```

```

pro save_variables_example

```

```

;create simple variables
sst_cccma_cgcm3_1 = 1
sst_cnrm_cm3 = 2
sst_csiro_mk3_0 = 3
sst_gfdl_cm2_0 = 4
sst_gfdl_cm2_1 = 5
sst_iap_fgoals1_0_g = 6
sst_inmcm3_0 = 7
sst_ipsl_cm4 = 8
sst_miroc3_2_medres = 9
sst_mpi_echam5 = 10
sst_mri_cgcm2_3_2a = 11
sst_ncar_ccsm3_0 = 12
sst_ncar_pcm1 = 13
sst_ukmo_hadcm3 = 14

```

```
sst_ukmo_hadgem1 = 15
```

```
sst = ['sst_cccma_cgcm3_1', 'sst_cnrm_cm3', 'sst_csiro_mk3_0', $  
'sst_gfdl_cm2_0', 'sst_gfdl_cm2_1', 'sst_iap_fgoals1_0_g', $  
'sst_inmcm3_0', 'sst_ipsl_cm4', 'sst_miroc3_2_medres', $  
'sst_mpi_echam5', 'sst_mri_cgcm2_3_2a', 'sst_ncar_ccsm3_0', $  
'sst_ncar_pcm1', 'sst_ukmo_hadcm3', 'sst_ukmo_hadgem1']
```

```
Result = Execute('SAVE, FILENAME="sst_cmip3_20c3m.sav", '+ $  
strJoin(sst, ', ', /Single) + ', /Verbose')
```

```
END
```

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [d.poreh](#) on Thu, 29 Sep 2011 13:26:31 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 29, 5:45 am, "Brian J. Daniel" <Daniels...@yahoo.com> wrote:

> On Sep 29, 8:03 am, Dave Poreh <d.po...@gmail.com> wrote:

>

>

>

>

>

>

>

>

>

>> On Sep 29, 4:50 am, "Brian J. Daniel" <Daniels...@yahoo.com> wrote:

>

>>> On Sep 29, 5:10 am, voidspace <jhkim...@gmail.com> wrote:

>

>>>> Hello folks,

>

>>>> I have a question at beginner's level. When I saved multiple variables

>>>> with same head but different tails in IDL, I found it is tedious to

>>>> type all of them in as follows.

>

>>>> SAVE, sst_cccma_cgcm3_1, sst_cnrm_cm3, sst_csiro_mk3_0,

>>>> sst_gfdl_cm2_0, sst_gfdl_cm2_1, \$

>>>> sst_iap_fgoals1_0_g, sst_inmcm3_0, sst_ipsl_cm4,

>>>> sst_miroc3_2_medres, sst_mpi_echam5, \$

>>>> sst_mri_cgcm2_3_2a, sst_ncar_ccsm3_0, sst_ncar_pcm1,

>>>> sst_ukmo_hadcm3, sst_ukmo_hadgem1, \$

>>>> FILENAME='sst_cmip3_20c3m.sav'

>

```

>>>> The '/All' keyword may help, but there are more variables that I don't
>>>> want to save in. This aroused my curiosity, so I post my question here
>>>> to know whether there is a workaround.
>
>>>> In fact, I tried to find an advanced way by myself, but found it's
>>>> difficult to know without an expert's guidance.
>
>>>> An attempt with my best knowledge is as follows. As all model names
>>>> from 'cccma_cgcm3_1' to 'ukmo_hadgem1' are saved in a string array
>>>> 'model', I considered the 'EXECUTE' command.
>
>>>> One prior step I did was saving 15 variable names in one single string
>>>> array 'sst'.
>
>>>> IDL> sst=STRARR(N_elements(model))
>>>> IDL> FOR i=0, N_elements(model)-1 DO sst[i]='sst_'+model[i]
>
>>>> Now, 'sst' contains 15 different variable names that I want to save
>>>> in. Then, tried the following
>
>>>> IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', "+sst[i] )
>
>>>> but in stuck because I have no idea how to make an implicit loop for
>>>> 'sst[i]'.
>
>>>> My attempt seems not a right way. I also thought it would be
>>>> convenient if there is implicit do-loop like (sst(i),i=1,15) in
>>>> Fortran, but I immediately got that it's nothing but a stupid idea.
>
>>>> I thought over and googled as well but all were in vain.
>
>>>> Can anybody suggest me a nice way to simplify the tedious command
>>>> aforementioned?
>
>>>> Best,
>>>> John
>
>>> You've already done the hard part, which is the string manipulation to
>>> create your 'sst' array. The final step is to use StrJoin to bring
>>> the array into one string joined by ', '. See below (UNTESTED).
>
>>> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,',
>>> ',/Single) )
>
>> Can you prepare a simple example? method seems quite good for me!
>> Cheers,
>> Dave
>

```

```

> pro save_variables_example
>
> ;create simple variables
> sst_cccma_cgcm3_1 = 1
> sst_cnrm_cm3 = 2
> sst_csiro_mk3_0 = 3
> sst_gfdl_cm2_0 = 4
> sst_gfdl_cm2_1 = 5
> sst_iap_fgoals1_0_g = 6
> sst_inmcm3_0 = 7
> sst_ipsl_cm4 = 8
> sst_miroc3_2_medres = 9
> sst_mpi_echam5 = 10
> sst_mri_cgcm2_3_2a = 11
> sst_ncar_ccsm3_0 = 12
> sst_ncar_pcm1 = 13
> sst_ukmo_hadcm3 = 14
> sst_ukmo_hadgem1 = 15
>
> sst = ['sst_cccma_cgcm3_1', 'sst_cnrm_cm3', 'sst_csiro_mk3_0', $
>       'sst_gfdl_cm2_0', 'sst_gfdl_cm2_1', 'sst_iap_fgoals1_0_g', $
>       'sst_inmcm3_0', 'sst_ipsl_cm4', 'sst_miroc3_2_medres', $
>       'sst_mpi_echam5', 'sst_mri_cgcm2_3_2a', 'sst_ncar_ccsm3_0', $
>       'sst_ncar_pcm1', 'sst_ukmo_hadcm3', 'sst_ukmo_hadgem1']
>
> Result = Execute('SAVE, FILENAME="sst_cmip3_20c3m.sav", '+ $
>       strJoin(sst, ', ', '/Single)+' ,/Verbose')
>
> END

```

Thanks Brian, it is very clever and for me works perfectly.

Cheers,

Dave

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [voidspace](#) on Thu, 29 Sep 2011 13:26:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

> On Sep 29, 8:03 am, Dave Poreh <d.po...@gmail.com> wrote:

>

>

>

>> On Sep 29, 4:50 am, "Brian J. Daniel" <Daniels...@yahoo.com> wrote:

>

>>> On Sep 29, 5:10 am, voidspace <jhkim...@gmail.com> wrote:


```

>
>>>> Hello folks,
>
>>>> I have a question at beginner's level. When I saved multiple variables
>>>> with same head but different tails in IDL, I found it is tedious to
>>>> type all of them in as follows.
>
>>>> SAVE, sst_cccma_cgcm3_1, sst_cnrm_cm3, sst_csiro_mk3_0,
>>>> sst_gfdl_cm2_0, sst_gfdl_cm2_1, $
>>>> sst_iap_fgoals1_0_g, sst_inmcm3_0, sst_ipsl_cm4,
>>>> sst_miroc3_2_medres, sst_mpi_echam5, $
>>>> sst_mri_cgcm2_3_2a, sst_ncar_ccsm3_0, sst_ncar_pcm1,
>>>> sst_ukmo_hadcm3, sst_ukmo_hadgem1, $
>>>> FILENAME='sst_cmip3_20c3m.sav'
>
>>>> The '/All' keyword may help, but there are more variables that I don't
>>>> want to save in. This aroused my curiosity, so I post my question here
>>>> to know whether there is a workaround.
>
>>>> In fact, I tried to find an advanced way by myself, but found it's
>>>> difficult to know without an expert's guidance.
>
>>>> An attempt with my best knowledge is as follows. As all model names
>>>> from 'cccma_cgcm3_1' to 'ukmo_hadgem1' are saved in a string array
>>>> 'model', I considered the 'EXECUTE' command.
>
>>>> One prior step I did was saving 15 variable names in one single string
>>>> array 'sst'.
>
>>>> IDL> sst=STRARR(N_elements(model))
>>>> IDL> FOR i=0, N_elements(model)-1 DO sst[i]='sst_'+model[i]
>
>>>> Now, 'sst' contains 15 different variable names that I want to save
>>>> in. Then, tried the following
>
>>>> IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', "+sst[i] )
>
>>>> but in stuck because I have no idea how to make an implicit loop for
>>>> 'sst[i]'.
>
>>>> My attempt seems not a right way. I also thought it would be
>>>> convenient if there is implicit do-loop like (sst(i),i=1,15) in
>>>> Fortran, but I immediately got that it's nothing but a stupid idea.
>
>>>> I thought over and googled as well but all were in vain.
>
>>>> Can anybody suggest me a nice way to simplify the tedious command
>>>> aforementioned?

```

```

>
>>>> Best,
>>>> John
>
>>> You've already done the hard part, which is the string manipulation to
>>> create your 'sst' array. The final step is to use StrJoin to bring
>>> the array into one string joined by ', '. See below (UNTESTED).
>
>>> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,',
>>> ',/Single) )
>
>> Can you prepare a simple example? method seems quite good for me!
>> Cheers,
>> Dave
>
> pro save_variables_example
>
> ;create simple variables
> sst_cccma_cgcm3_1 = 1
> sst_cnrm_cm3 = 2
> sst_csiro_mk3_0 = 3
> sst_gfdl_cm2_0 = 4
> sst_gfdl_cm2_1 = 5
> sst_iap_fgoals1_0_g = 6
> sst_inmcm3_0 = 7
> sst_ipsl_cm4 = 8
> sst_miroc3_2_medres = 9
> sst_mpi_echam5 = 10
> sst_mri_cgcm2_3_2a = 11
> sst_ncar_ccsm3_0 = 12
> sst_ncar_pcm1 = 13
> sst_ukmo_hadcm3 = 14
> sst_ukmo_hadgem1 = 15
>
> sst = ['sst_cccma_cgcm3_1', 'sst_cnrm_cm3', 'sst_csiro_mk3_0', $
>        'sst_gfdl_cm2_0', 'sst_gfdl_cm2_1', 'sst_iap_fgoals1_0_g', $
>        'sst_inmcm3_0', 'sst_ipsl_cm4', 'sst_miroc3_2_medres', $
>        'sst_mpi_echam5', 'sst_mri_cgcm2_3_2a', 'sst_ncar_ccsm3_0', $
>        'sst_ncar_pcm1', 'sst_ukmo_hadcm3', 'sst_ukmo_hadgem1']
>
> Result = Execute('SAVE, FILENAME="sst_cmip3_20c3m.sav", '+ $
>        strJoin(sst, ', ', '/Single)+' ,/Verbose')
>
> END

```

Thanks again for this good example!

I need to look through the powerful functions related to string manipulation.

Best,
John

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [voidspace](#) on Thu, 29 Sep 2011 13:34:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

This is my test for an example for Dave..

Dear Brian (and Dave for an example),

It really worked with a minor bug fixing! Really simple and powerful function that I didn't know.

I tried printing first and got the variable names concatenated by comma that I wanted to produce.

```
IDL> print,strJoin(sst,',',/Single)
```

```
sst_cccma_cgcm3_1,sst_cnrm_cm3,sst_csiro_mk3_0,sst_gfdl_cm2_
0,sst_gfdl_cm2_1,sst_iap_fgoals1_0_g,sst_inmcm3_0,sst_ipsl_c
m4,sst_miroc3_2_medres,sst_mpi_echam5,sst_mri_cgcm2_3_2a,sst
_ncar_ccsm3_0,sst_ncar_pcm1,sst_ukmo_hadcm3,sst_ukmo_hadgem1
```

Then, I tried first without 'EXECUTE' function because Brian's suggestion doesn't look require it.

```
IDL> SAVE, FILENAME='sst_cmip3_20c3m.sav', strJoin(sst,',',/Single)
% SAVE: Expression must be named variable in this context: <STRING
('sst_cccma_cgcm3_1,ss...')>.
```

It didn't work! So I tried with 'EXECUTE' function following Brian's suggestion, but got the same error message.

```
IDL> result=EXECUTE("SAVE, FILENAME='sst_cmip3_20c3m.sav',
strJoin(sst,',',/Single)")
% SAVE: Expression must be named variable in this context: <STRING
('sst_cccma_cgcm3_1,ss...')>.
```

Then, I thought that strJoin(sst,',',/Single) part should be outside of quotation marks according to the error message.

Tried the following and got exactly what I wanted!

```
IDL> result=EXECUTE("SAVE,  
FILENAME='sst_cmip3_20c3m.sav','"+strJoin(sst,',',/Single))
```

Thank you very much for your help, Brian!

Cheers,
John

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [greg.addr](#) on Thu, 29 Sep 2011 13:50:44 GMT

[View Forum Message](#) <> [Reply to Message](#)

Although you can use tricks like this to do what you need, you would almost certainly have been better off to define your variables as arrays in the first place, and refer to them by their indices...

e.g.
sst_cccma_cgcm3[i]

Otherwise, you'll meet the same problem every time you want to do anything at all with their values (plot them, for example)

good luck!
Greg

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [voidspace](#) on Thu, 29 Sep 2011 14:16:50 GMT

[View Forum Message](#) <> [Reply to Message](#)

> Although you can use tricks like this to do what you need, you would almost certainly have been better off to define your variables as arrays in the first place, and refer to them by their indices...

>

> e.g.

> sst_cccma_cgcm3[i]

>

> Otherwise, you'll meet the same problem every time you want to do anything at all with their values (plot them, for example)

>

> good luck!

> Greg

Thanks, Greg. Exactly. In fact I could put all different variables

into one defined variable and save all of them in one array variable. This is certainly much better way. But today I had a good exercise about the string manipulation in IDL. Cheers, John

Subject: Re: Is there a work around to save multiple variables with same head but different tails without listing all of them?

Posted by [Brian Daniel](#) on Thu, 29 Sep 2011 14:46:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Sep 29, 10:16 am, voidspace <jhkim...@gmail.com> wrote:

>
>> Although you can use tricks like this to do what you need, you would almost certainly have been better off to define your variables as arrays in the first place, and refer to them by their indices...
>
>> e.g.
>> sst_cccma_cgcm3[i]
>
>> Otherwise, you'll meet the same problem every time you want to do anything at all with their values (plot them, for example)
>
>> good luck!
>> Greg
>
> Thanks, Greg. Exactly. In fact I could put all different variables
> into one defined variable and save all of them in one array variable.
> This is certainly much better way. But today I had a good exercise
> about the string manipulation in IDL. Cheers, John

My guess is the poster's variables are arrays with varying size. I used integers for simplicity in the example. In the case of arrays, you're better off using a structure of arrays.
