

---

Subject: Re: Substituting multiple strings into a program  
Posted by [Paul Van Delst\[1\]](#) on Wed, 08 Aug 2012 20:16:20 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

So you want to read a "control" file containing a list of filenames (or portions of filenames), and then loop over that filename list reading and displaying the data in each file?

On 08/08/12 15:50, MightyMrFish wrote:

- > I believe the topic may be a bit confusing, so I'll try to describe what I wish to do here. You'll have to forgive me
- > if I don't use the correct terminology as I'm not exactly skilled in any sort of programming.
- >
- > What I want to do is get individual strings of text from a file and place them into a program, run that program, and
- > then repeat the program with the next string of text in the file. For example, say I have a file titled "DogNames"
- > that has a list of names of dogs (Rex, Cherry, Sniffles, Captain Kevin the V, etc.). I have a set of text documents
- > with data corresponding to each name. The text documents are titled according to which dog the data is from (data on
- > Rex is titled Rex, data from Cherry is titled Cherry, etc.) The data is columns of information. I have a program
- > already that calls for the user to input the name of a file to draw data from and what name to give the graph it
- > spits out.
- >
- > I'd like to be able to have the program read a list of names and, since they directly correspond to file names, be
- > able to sub in each name into the program I already wrote and spit out a graph with the same name. I can do the
- > latter (naming the graphs), but the former is what I'm unsure about. I know what I want to do (read name1, insert
- > into program, produce graph, read name2, insert into program, etc...), but don't know what commands to use or even
- > the appropriate syntax.
- >
- > Any advice/help would be greatly appreciated.

---

Subject: Re: Substituting multiple strings into a program  
Posted by [MightyMrFish](#) on Wed, 08 Aug 2012 21:04:12 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I want something that will read one of the filenames and insert it into my program.  
How I have it set up right now, I'm using this command line:  
name ="

read, 'What is the filename?', name

Then I use a READCOL command to plot the data:

READCOL, name, ...

I'd like it if I could have the "name" part automatically filled in with the names on the list. One by one of course. Once the name is inserted, the rest of the command does its thing and I get a graph of the data.

If it helps, I'll post my program.

On Wednesday, August 8, 2012 1:16:20 PM UTC-7, Paul van Delst wrote:

> So you want to read a "control" file containing a list of filenames (or portions of filenames), and then loop over that  
>  
> filename list reading and displaying the data in each file?

---

---

Subject: Re: Substituting multiple strings into a program  
Posted by [Paul Van Delst\[1\]](#) on Wed, 08 Aug 2012 21:49:02 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On 08/08/12 17:04, MightyMrFish wrote:

> I want something that will read one of the filenames and insert it into my program. How I have it set up right now,  
> I'm using this command line: name =" read, 'What is the filename?', name  
>  
> Then I use a READCOL command to plot the data:  
>  
> READCOL, name, i½  
>  
> I'd like it if I could have the "name" part automatically filled in with the names on the list. One by one of course.  
> Once the name is inserted, the rest of the command does its thing and I get a graph of the data.  
>  
> If it helps, I'll post my program.  
>  
>  
> On Wednesday, August 8, 2012 1:16:20 PM UTC-7, Paul van Delst wrote:  
>> So you want to read a "control" file containing a list of filenames (or portions of filenames), and then loop over  
>> that  
>>  
>> filename list reading and displaying the data in each file?

O.k., still not sure I understand. But, here's what I came up with:

1) First create (what I call) the control file:

```
$ cat > dognames.txt
Rex
Cherry
Sniffles
CaptainKevin
^D
```

2) O.k., now the following procedure reads that file

```
PRO ReadControlFile, control_file

; Open the control file
OPENR, fid, control_file, /GET_LUN

; Loop over the entries in the control file
entry = ""
WHILE ~EOF(fid) DO BEGIN
    READF, fid, entry
    PRINT, "The current entry is: ", entry
ENDWHILE

; Close the control file
FREE_LUN, fid

END
```

```
IDL> .run readcontrolfile
% Compiled module: READCONTROLFILE.
IDL> readcontrolfile, 'dognames.txt'
The current entry is: Rex
The current entry is: Cherry
The current entry is: Sniffles
The current entry is: CaptainKevin
```

3) Now let's assume there is a separate datafile associated with each entry in the control file, e.g.

```
$ cat Rex.dat
# time woof-factor
1 2.4
2 3.6
3 4.8
4 15.6
5 6.7
6 5.2
```

Let's just copy the one file for the other doggies,

```
$ cp Rex.dat Cherry.dat
$ cp Rex.dat Sniffles.dat
$ cp Rex.dat CaptainKevin.dat
```

and edit them so they have different numbers of lines in them:

```
$ wc -l *.dat
4 CaptainKevin.dat
6 Cherry.dat
7 Rex.dat
5 Sniffles.dat
22 total
```

4) Let's write a script to read that sort of datafile and return its data (doing this bit simply is probably the hardest part),

```
PRO ReadDataFile, data_file, x, y

; How many lines?
n_lines = FILE_LINES(data_file)

; Assume first line is comment
n_points = n_lines - 1

; And now create the output x and y arrays
x = FLTARR(n_points)
y = FLTARR(n_points)

; Open the data file
OPENR, fid, data_file, /GET_LUN

; Loop over the lines in the datafile
entry = "
a = 0.0
```

```

b = 0.0
FOR i = -1, n_points - 1 DO BEGIN
    READF, fid, entry          ; Read the line into a string
    IF ( i EQ -1 ) THEN CONTINUE ; If it's the comment line, skip it
    READS, entry, a, b          ; Read the data from the string
    x[i] = a                    ; Assign the data to...
    y[i] = b                    ; ...the output arrays
ENDFOR

; Close the data file
FREE_LUN, fid

END

```

```

IDL> .run readcontrolfile
% Compiled module: READDATAFILE.
% Compiled module: READCONTROLFILE.
IDL> readdatafile, 'Rex.dat', x, y
IDL> help, x, y
X          FLOAT    = Array[6]
Y          FLOAT    = Array[6]
IDL> print, x, y
  1.00000   2.00000   3.00000   4.00000   5.00000   6.00000
  2.40000   3.60000   4.80000  15.6000   6.70000   5.20000

```

5) Now modify the control file reader to also read the individual data files:

```

PRO ReadControlFile, control_file

; Open the control file
OPENR, fid, control_file, /GET_LUN

; Loop over the entries in the control file
entry = ""
WHILE ~EOF(fid) DO BEGIN
    READF, fid, entry
    PRINT, "The current entry is: ", entry, ". Reading the associated datafile..."
    ReadDataFile, entry+'.dat', x, y
    HELP, x, y
ENDWHILE

; Close the control file
FREE_LUN, fid

END

```

```

IDL> .run readcontrolfile
% Compiled module: READDATAFILE.
% Compiled module: READCONTROLFILE.
IDL> readcontrolfile,'dognames.txt'
The current entry is: Rex. Reading the associated datafile...
X      FLOAT    = Array[6]
Y      FLOAT    = Array[6]
The current entry is: Cherry. Reading the associated datafile...
X      FLOAT    = Array[5]
Y      FLOAT    = Array[5]
The current entry is: Sniffles. Reading the associated datafile...
X      FLOAT    = Array[4]
Y      FLOAT    = Array[4]
The current entry is: CaptainKevin. Reading the associated datafile...
X      FLOAT    = Array[3]
Y      FLOAT    = Array[3]

```

Of course, rather than "HELP, x, y" you can do "PLOT, x, y" or whatever.

cheers,

paulv

---

Subject: Re: Substituting multiple strings into a program  
 Posted by [MightyMrFish](#) on Thu, 09 Aug 2012 19:37:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

You'll have to forgive the lateness of my reply. I have access to the IDL programming software at certain times of the day.

I tried using that code, but issues arose. I couldn't get the graphs to be correct.

I think posting the code I'm using right now will help me explain what I want to do. It uses the READCOL.pro which can be found here: <http://idlastro.gsfc.nasa.gov/ftp/pro/misc/readcol.pro>

CD, '[FOLDER]'; Replace [FOLDER] with the folder containing the data tables.

name ="

read, 'What is the filename to plot? (Write exactly as it appears.) ',name

; Defining "name" for the READCOL command. The user inputs the name of the

; data table file (so let's say Rex.dat just to be consistant).

READCOL, name, W1, W1Sigma, W2, W2Sigma, W3, W3Sigma, W4, W4Sigma, MJD, FORMAT  
 = 'X,X,X,X,X,X,X,X, D, D, D, D, D, D, D, D, D'

; The W1, W1Sigma, etc. refer to columns of data. The data I'm really using is

; from the WISE obervatory. The W's correspond to wavelengths.

!P.MULTI = [0,0,4,0,1] ; This makes it so there are 4 graphs on a "page".

```
PLOTERROR, MJD, W1, W1Sigma, psym=1, $
TITLE = 'W1', XTITLE = 'MJD', YTITLE = 'Mag.'
WAIT, .5
PLOTERROR, MJD, W2, W2Sigma, psym=1, $
TITLE = 'W2', XTITLE = 'MJD', YTITLE = 'Mag.'
WAIT, .5
PLOTERROR, MJD, W3, W3Sigma, psym=1, $
TITLE = 'W3', XTITLE = 'MJD', YTITLE = 'Mag.'
WAIT, .5
PLOTERROR, MJD, W4, W4Sigma, psym=1, $
TITLE = 'W4', XTITLE = 'MJD', YTITLE = 'Mag.'
WAIT, .5
; The WAIT isn't needed. The PLOTERROR command gives a plot
; with error bars.
```

```
set_plot, "ps"
set_plot, "x"
```

```
CD, '..'
CD, '[OUTPUT FOLDER]'
; Replace [OUTPUT FOLDER] with the name of the folder you want
; the graphs to come out of.
```

```
filename='~user/name'; Replace 'user' accordingly.
write_png, name, tvrd()
```

```
CD, '..'
```

```
END
```

This .pro reads the data table file and spits out graphs according to what I need to be graphed. What I'd like is some way for me to not have to enter the names of each file. I have a list of the data files. I want to be able to have my .pro read one name off the list, assign it to "name", go through the command I've written to spit out a graph, then repeat with the next name on the list.

---