

IDL Tutorials

Matt Johnson

April 3, 2002

1 Tutorial One

We are interested in using IDL to manipulate and visualize data. Data in IDL is in the form of multidimensional arrays (matrices). For instance, we may create an array that represents the value of $\sin(x)$ at different x by doing the following:

1. First, define the variable x as an array:

```
x=2*!pi/100*findgen(100)
```

Note that π is typed as `!pi` in IDL. The `findgen()` command creates an array composed of integers starting at zero up to the number indicated in the parentheses. We multiplied by $2\pi/100$ to create an array consisting of 100 evenly spaced data points between 0 and 2π . To view the elements of x , simply type:

```
print, x
```

2. Now, we define a new array, y , as the value of $\sin(x)$ at each of our 100 evenly spaced data points:

```
y=sin(x)
```

3. So, if we want to graph x vs y , we simply type:

```
plot, x, y
```

The independent variable comes first, then the dependent. The plot should appear as it does in figure 1.

There are additional options that can be added to the `plot` command:

```
plot, x, y, title='sin(x)', xtitle='x', ytitle='sin(x)'
```

This creates a title for the plot and labels the axes. Note the command structure of IDL. First the main command is typed, and all variables and options are separated by commas. This graph should look like figure 2.

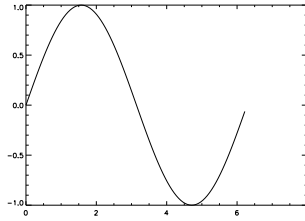


Figure 1:

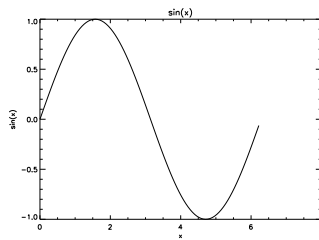


Figure 2:

2 Tutorial Two

To create a post script (graphics) file of the graph that we just made, we must change the output of the *plot* command. This is accomplished by the *setplot* command:

1. Set the output of the *plot* command to post script:

```
set_plot, 'ps'
```

2. Specify the name you wish the file to have:

```
device, filename='name.ps'
```

3. Now, give the *plot* command:

```
plot, x, y
```

4. In order to save the file, you must close the device:

```
device, /close
```

5. To set the plot command back to the screen, just type:

```
set_plot, 'win'
```

If you are running a unix based platform, you would type:

```
set_plot, 'x'
```

3 Tutorial Three

Thus far, we have been typing in a series of commands at the IDL prompt. However, you may write series of commands into a text file and have IDL read this. Let's write a program!

1. In IDL, go to 'File', 'New', 'Editor' to start a new file into which you may type your program. You may also use any other ASCII text editor like Notepad. Always save your files as *.pro* so that IDL is able to recognize them as programs.
2. Now, type the string of commands needed to create a post script file of your $\sin(x)$ graph:

```
set_plot, 'ps'  
device, filename='name.ps'  
plot, x, y  
device, /close  
set_plot, 'win'  
end
```

Note that we must tell IDL when the program is over by typing *end*.

3. Save your program as a *.pro* file, now we are ready to run it!
4. To run your program, you must first compile it into the computer's memory using the *.run* command:

```
.run name.pro
```

Now, to execute the program we use the *.go* command:

```
.go
```

The program should have created a post script file of your $\sin(x)$ graph!