

For practice, start by creating a journal for the session: this is Tutorial 4.

String Manipulation

We want to extract information from strings. Create the following variables (if you want, you can copy and paste them from `/home/shared/astr2600/data/tutorial3_data.txt` - use `cat` to see its contents):

```
IDL> R = "Rigel          magB=  0.9 magV= 0.12"
IDL> S = "Sirius         magB=-1.46 magV=-1.46"
IDL> V = "Vega           magB= 0.03 magV= 0.03"
IDL> B = "Betelgeuse    magB= 2.27 magV= 0.42"
```

Our goal is to extract B and V magnitudes for these stars and enter them into arrays.

First, find out where the `magB` value is stored. To do this, let's look for the string `"magB="`.

```
IDL> magBstart = strpos(R,"magB=")
IDL> print,magBstart
```

If you've done this correctly, `magBstart` should be 11. OK, great, but we don't care where the string `"magB"` starts, we care where it *ends*. So, let's use `strlen`:

```
IDL> magBlen = strlen("magB=")
IDL> magBend = magBstart + magBlen
```

All good so far. Let's do the same for `magV`:

```
IDL> magVstart = strpos(R,"magV=")
IDL> magVlen = strlen("magV=")
IDL> magVend = magVstart + magVlen
```

Do a sanity check: What are the values of `magVstart` etc? Do they make sense?

Now we actually want to extract the magnitudes. We'll use `strmid` for that purpose.

```
IDL> magB = strmid(R, magBend, magBlen)
IDL> magV = strmid(R, magVend, magVlen)
IDL> print,"Rigel magB is ",magB," and magV is ",magV
```

Great, we've read one of the strings! This whole process was pretty tedious, so we ideally don't want to have to re-do it. We can note that the strings were all of the same length:

```
IDL> print,strlen(B)
IDL> print,strlen(V)
IDL> print,strlen(S)
IDL> print,strlen(R)
```

and that they all have `magB` and `magV` in the same places. So all we need to do is the `strmid` step, not the others.

Since we're making arrays, we'll do this all as one big step...

```
IDL> magB = [strmid(R, magBend, magBlen),strmid(B, magBend,
           magBlen),strmid(S, magBend, magBlen),strmid(V, magBend, magBlen)]
IDL> magV = [strmid(R, magVend, magVlen),strmid(B, magVend,
           magVlen),strmid(S, magVend, magVlen),strmid(V, magVend, magVlen)]
IDL> print,magB
IDL> print,magV
IDL> help,magB,magV
```

But these are still of type string! Let's make them floats:

```
IDL> magB = float(magB)
IDL> magV = float(magV)
```

Now, finally, we have some arrays with real data in them. We'll make a color-magnitude diagram using IDL's plot routine. Try each of these commands in order, they'll look slightly different:

```
IDL> ; this first one will look a bit strange
IDL> plot,magB-magV,magV
IDL> ; this is a little better
IDL> plot,magB-magV,magV,psym=1
IDL> ; maybe stars are best
IDL> plot,magB-magV,magV,psym=2
IDL> ; Make it colorful
IDL> plot,magB-magV,magV,psym=2
IDL> oplot,magB-magV,magV,psym=2,color='0000FF'x,symsize=10
```

OK, those plots didn't look great - not many data points. Let's load more data and try again. As above, copy & paste lines from `/home/shared/astr2600/data/tutorial3_data.txt`:

```
IDL> pleiadesB = [2.806, 3.54, 3.612, 3.812, 4.113, 4.199, 4.967, 5.406,
5.585, 5.727, 6.124, 6.30, 6.415, 6.586, 6.87, 6.85]
IDL> pleiadesV = [2.873, 3.62, 3.705, 3.871, 4.164, 4.291, 5.048, 5.448,
5.651, 5.761, 6.172, 6.28, 6.430, 6.606, 6.81, 6.83]
```

Then, let's try the same plots as above:

```
IDL> ; this first one will look a bit strange
IDL> plot,pleiadesB-pleiadesV,pleiadesV
IDL> ; this is a little better
IDL> plot,pleiadesB-pleiadesV,pleiadesV,psym=1
IDL> ; maybe stars are best
IDL> plot,pleiadesB-pleiadesV,pleiadesV,psym=2
IDL> ; Make it colorful
IDL> ; note the single quotes (not double) in this line
IDL> oplot,pleiadesB-pleiadesV,pleiadesV,psym=2,color='0000FF'x,symsize=3
```

To give you some idea of the astrophysics involved, plus get you used to labeling things, we'll add a title and axis labels

```
IDL> plot,pleiadesB-pleiadesV,pleiadesV,psym=2,xtitle='B-V',ytitle='V
magnitude',title='Pleiades Color-Magnitude Diagram'
IDL> oplot,pleiadesB-pleiadesV,pleiadesV,psym=2,color='0000FF'x,symsize=3
```

Finally, for a real color-magnitude diagram, we want the brightest things plotted at the top.

```
IDL> plot,pleiadesB-pleiadesV,pleiadesV,psym=2,xtitle='B-V',ytitle='V
magnitude',title='Pleiades Color-Magnitude Diagram',yrange=[8,2]
IDL> oplot,pleiadesB-pleiadesV,pleiadesV,psym=2,color='0000FF'x,symsize=3
```

Color-magnitude diagrams are useful plots for determining properties of star clusters and galaxies. The y-axis, the brightness of the source, is strongly dependent on distance and the star's luminosity, while the x-axis depends on the star's intrinsic properties, but not distance. When looking at a cluster or a galaxy, all stars are at (approximately) the same distance, and therefore the y-axis only depends on the luminosity. The Pleiades cluster is one of the nearest and most massive clusters we have complete information on (meaning we think we know its distance very accurately), so it serves as a calibration target for stellar properties.

Python

String manipulation in IDL is tedious in ugly, but in python it can be a little nicer.

Start up your ipython terminal with `--pylab` (it should create an `ipython_log` file automatically) and enter the same text as before:

```
>>> R = "Rigel          magB=  0.9 magV=  0.12"
>>> S = "Sirius         magB=-1.46 magV=-1.46"
>>> V = "Vega           magB=  0.03 magV=  0.03"
>>> B = "Betelgeuse     magB=  2.27 magV=  0.42"
```

Then the equivalent search commands:

```
>>> magBstart = B.find("magB=")
>>> magBlen = len("magB=")
>>> magBend = magBstart + magBlen
>>> magVstart = B.find("magV=")
>>> magVlen = len("magV=")
>>> magVend = magVstart + magVlen
```

In python, you can treat arrays like lists.

```
>>> magB = float( B[magBend:magVstart] )
>>> magV = float( B[magVend:] )
```

Now, a nice trick to get all magnitudes. This is called a ‘list comprehension’ and is somewhat complicated, so we’ll cover it again later, but for now you can just enter these commands:

```
>>> Bmags = [float(x[magBend:magVstart]) for x in [R,S,V,B]]
>>> Vmags = [float(x[magVend:]) for x in [R,S,V,B]]
```

Finally, plot the data:

```
>>> plot(Bmags ,Vmags , 'o')
```