Subject: Re: How does one read in simple binary files?
Posted by thompson on Sat, 13 Jan 1996 08:00:00 GMT
View Forum Message <> Reply to Message

bmac@igpp.llnl.gov (Bruce Macintosh) writes:

> This is an incredibly basic question: what's the simplest way
> to read in a simple binary file (ie an file of 16384 2-byte integers
> representing a 128x128 pixel image, for example), or slightly
> more complicated binary files (an array of 16384 IEEE 32-bit reals
> with a 100-byte header preceeding the pixel values, for example.)
> Are there any standard packages/routines for handling this kind of i/o,
> with or without byte and word swapping? All
> I can find in the manuals are routines for reading in ascii text, or
> various specialized formats, but nothing generic. Does "unformatted i/o"
> (readu, etc.) do this sort of input?

Yep.

All you would need to do for your example of a 128x128 array would be the following:

```
IDL> OPENR, UNIT, 'filename1', /GET_LUN
IDL> I_ARRAY = INTARR(128,128)
IDL> READU, UNIT, I_ARRAY
IDL> FREE_LUN, UNIT
```

or for your more complicated example

```
IDL> OPENR, UNIT, 'filename2', /GET_LUN
IDL> HEADER = BYTARR(100)
IDL> READU, UNIT, HEADER
IDL> F_ARRAY = FLTARR(16384) ;Or did you mean (128,128) again?
IDL> READU, UNIT, F_ARRAY
IDL> FREE_LUN, UNIT
```

You can also use ASSOC to do the same thing--check the documentation on the use of ASSOC.

If the data is written in a binary format different from that of the host computer, then it's a little more complicated. The BYTEORDER routine can be used to convert between standard and host-specific byte ordering, and even between IEEE and host floating point representations. The above two examples might also include the following lines after the read statements.

```
IDL> BYTEORDER, I_ARRAY, /NTOHS

IDL> BYTEORDER, F_ARRAY, /XDRTOF
```

Personally, I recommend using the routines IEEE_TO_HOST and HOST_TO_IEEE from the Astronomy User's Library. That way, you don't have to figure out what the proper keyword to use with BYTEORDER.

William Thompson