Subject: Re: Strange arrary indexing quirk?
Posted by penteado on Fri, 12 Aug 2011 03:42:32 GMT
View Forum Message <> Reply to Message

On Aug 11, 10:12 pm, Beaker <mattjamesfran...@gmail.com> wrote:
> Does anyone have any insight into why this happens? I'd expect the
> second line to also throw an error. A lot of the inner workings of IDL
> seem pretty arcane and hard to guess at before you try things...
>
> I was being lazy/naughty and using a catch statement to detect when an
> index went out of range, which I knew would happen very occaisonally
> and was trying to avoid the performance hit of checking for it every
> time. This round off instead of an error behaviour made sure that plan
> didn't work!

If you do not like that behaviour, use

compile_opt strictarraysubs

Subject: Re: Strange arrary indexing quirk?
Posted by penteado on Fri, 12 Aug 2011 03:45:56 GMT
View Forum Message <> Reply to Message

On Aug 11, 10:12 pm, Beaker <mattjamesfran...@gmail.com> wrote:
> Does anyone have any insight into why this happens? I'd expect the
> second line to also throw an error. A lot of the inner workings of IDL
> seem pretty arcane and hard to guess at before you try things...

It happens because it was chosen that way when the language was
defined. Quoting from the help:

"STRICTARRSUBS — When IDL subscripts one array using another array as the source of array indices, the default behavior is to clip any out-of-range indices into range and then quietly use the resulting data without error. This behavior is described in Understanding Array Subscripts. Specifying STRICTARRSUBS will instead cause IDL to treat such out-of-range array subscripts within the body of the routine containing the COMPILE_OPT statement as an error. The position of the STRICTARRSUBS option within the module is not important: All subscripting operations within the entire body of the specified routine will be treated this way."

Subject: Re: Strange arrary indexing quirk?
Posted by David Fanning on Fri, 12 Aug 2011 03:58:43 GMT
View Forum Message <> Reply to Message

Beaker writes:

> I stumbled across this strange behaviour:
>
> foo = indgen(2,2)
> print,foo[0,10]
> < gives an error, since the second index is out of range>
>
> however,
>
> indx = indgen(5)
> print,foo[0,indx]
> 0
> 2
> 2
> 2
> 2
>
> Does anyone have any insight into why this happens? I'd expect the
> second line to also throw an error. A lot of the inner workings of IDL
> seem pretty arcane and hard to guess at before you try things...
>
> I was being lazy/naughty and using a catch statement to detect when an
> index went out of range, which I knew would happen very occaisonally
> and was trying to avoid the performance hit of checking for it every
> time. This round off instead of an error behaviour made sure that plan
> didn't work!

This is a famous IDL quirk. If you would prefer to have
an error thrown in this situation, use the compile option
STRICTARRSUBS:

```
Compile_Opt STRICTARRSUBS
```

Looking up that compiler option will also point you to information
about array subscripts. (At least it will in the IDL 7 on-line help.)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Strange arrary indexing quirk?
Posted by Matt Francis on Fri, 12 Aug 2011 05:05:36 GMT
View Forum Message <> Reply to Message

Thanks for the help. That's a real doozy!