
Subject: Re: FOR loop ends one above where it's supposed to
Posted by [mankoff](#) on Wed, 02 Jun 2010 00:53:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jun 1, 5:26 pm, Bulgakovv <alex.ph.sjog...@gmail.com> wrote:

```
> Maybe there are previous posts about this, but I tried to google
> around without any success so I'll throw the question out.
>
> When I make a FOR loop from for example 0 to 5, the counter ends on
> the value one step above what I stated. Here's an example:
>
>> for i =0,5 do print,i
>
>      0
>      1
>      2
>      3
>      4
>      5>print,i
>
>      6
>
> Is this how it's supposed to be in IDL? To me it seems a little bit
> weird, why would the FOR loop count one extra step in the end? Took me
> a couple of hours to realize what was going on in my code as I took
> for granted this was not the case, so now I'm interested in
> understanding why IDL is behaving like this. Thanks and sorry if
> reposting!
```

The way I think of for loops is this:

```
>> for i =0,5 do print,i
1) Set i = 0
2) Check if i gt 5
3) If not, do the loop
4) increment i
5) Go to step (2)
```

So, step 4 happens incrementing from 5 to 6 at the end of the loop, and then it continues past the end of the loop, but no loop events occur with i equal to 6. Every other language I have used behaves analogous to this, I think.

-k.
