Subject: Re: pred & succ ?
Posted by Martin Schultz on Mon, 13 Nov 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Andrew wrote:
>
> In article <3A0FA91A.7BEE1D07@dkrz.de>,
>   Martin Schultz <martin.schultz@dkrz.de> wrote:
>>
>> FUNCTION pred, x
>>   RETURN, x-1
>> END
>> FUNCTION succ, x
>>   RETURN, x+1
>> END
>>
>
> Thanks, Martin. I take that as a "no", though.
>
> I was thinking of a general, built-in (thus fast),
> function that returns the successor or predecessor
> of any ordinal variable.
>
> I knew I could write x-1, etc.. :-)
>

Well, the point is that x-1 already is quite fast because IDL is array
oriented. I'm not sure whether you could gain anything significant by
having this as a builtin function.

Cheers,
Martin

--
[[[[[[[[[[
[[Dr. Martin Schultz   Max-Planck-Institut fuer Meteorologie   [[
[[Bundesstr. 55, 20146 Hamburg   [[
[[phone: +49 40 41173-308   [[
[[fax: +49 40 41173-298   [[
[[martin.schultz@dkrz.de   [[
[[[[[[[[[[

Subject: Re: pred & succ ?
Posted by noymer on Mon, 13 Nov 2000 08:00:00 GMT

View Forum Message <> Reply to Message

In article <3A0FA91A.7BEE1D07@dkrz.de>,
  Martin Schultz <martin.schultz@dkrz.de> wrote:
>
> FUNCTION pred, x
>   RETURN, x-1
> END
> FUNCTION succ, x
>   RETURN, x+1
> END
>

Thanks, Martin. I take that as a "no", though.

I was thinking of a general, built-in (thus fast),
function that returns the successor or predecessor
of any ordinal variable.

I knew I could write x-1, etc.. :-)

-- Andrew

Subject: Re: pred & succ ?
Posted by Martin Schultz on Mon, 13 Nov 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Andrew wrote:
>
> Dear C.I.i-p,
>
>   In Pascal (and probably in other high-level languages,
> but I don't know), there are built-in functions PRED(x)
> & SUCC(x), where x is any ordinal-type variable.
>
> e.g.: PRED(1)=0
>   SUCC(2)=3
>   PRED(TRUE)=FALSE
>
> etc., etc.
>
>   Is there anything like this in IDL?
>

> -- Andrew
>
> Sent via Deja.com http://www.deja.com/
> Before you buy.

```
FUNCTION pred, x
  RETURN, x-1
END
FUNCTION succ, x
  RETURN, x+1
END
```

But is that really worth it? You won't be able to construct
pred(false) for there is no distinct "false" in IDL. But this is
actually a nice little "Denksportaufgabe" (although I am sure you can
find a solution on some webpage, too): How do you construct a reliable
complement of a variable that represents a boolean?
Hints:
- should accept any (numerical) type (ok, maybe not complex ;-)
- variable type should be preserved
- 0 is false, everything else true (how about 1.e-30? - probably best
to define an "eps"); in output, "true" should be 1
- should not be constrained to scalars (IF(...) won't work)

Thanks to RSI for not providing us with a dedicated "logical" or
"boolean" type. Otherwise this would be too trivial ;-)

Cheers,
Martin

--
[[[[[[[[[[
[[Dr. Martin Schultz   Max-Planck-Institut fuer Meteorologie   [[
[[Bundesstr. 55, 20146 Hamburg   [[
[[phone: +49 40 41173-308   [[
[[fax: +49 40 41173-298   [[
[[martin.schultz@dkrz.de   [[
[[[[[[[[[[