

---

Subject: Re: random numbers in version 4.0

Posted by [Jack Saba](#) on Wed, 26 Nov 1997 08:00:00 GMT

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

---

RANDOMN uses the Box-Muller algorithm. This generates two random numbers each time. In the FORTRAN implementation in Numerical Recipes, calls #2,4,... to the subroutine return the "extra" value generated on the previous call, and do NOT change the seed. I expect the same thing is happening with the IDL implementation.

The only way around this appears to make sure you make an even number of calls before beginning your next sequence.

Kirt Schaper wrote:

```
>
> Has anyone else encountered odd behavior with random numbers in
> IDL version 4.0? Specifically, if you repeatedly generate an odd
> number of random numbers with the same seed, you don't get the same
> numbers (strictly speaking, you do get the same numbers, but shifted
> one position). For example:
>
> % idl
> IDL. Version 4.0.1 (sunos sparc).
> Copyright 1989-1995, Research Systems, Inc.
> All rights reserved. Unauthorized reproduction prohibited.
> Installation number: XXXXXXXX.
> Licensed for use by: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
>
> Initial setup: quiet=1, order=0
> IDL> seed = 10.
> IDL> s = seed & print,randomn(s,3)
> 2.10917 1.36675 -1.02318
> IDL> s = seed & print,randomn(s,3)
> -0.568035 2.10917 1.36675
> IDL> s = seed & print,randomn(s,3)
> 2.10917 1.36675 -1.02318
> IDL> s = seed & print,randomn(s,3)
> -0.568035 2.10917 1.36675
>
> This behavior has been repeated on HPUNIX, Digital UNIX and Solaris
> version of IDL. It seems to have gone away in version 5 but we
> are stuck with version 4.0 until the convolv 2X slowdown in
> version 5 is fixed.
>
> kt
```

--

Jack Saba <jack@magus.stx.com>

---