
Subject: Re: Randomu() behavior - BAD!

Posted by [David Fanning](#) on Fri, 13 Nov 2009 19:26:23 GMT

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

Conor writes:

> So I did a simple test and have a better idea of how the seed
> initialization works with randomu().

Most interesting. What version of IDL are you using, Conor?

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Randomu() behavior - BAD!

Posted by [Conor](#) on Fri, 13 Nov 2009 20:10:52 GMT

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

On Nov 13, 2:26 pm, David Fanning <n...@dfanning.com> wrote:

> Conor writes:

>> So I did a simple test and have a better idea of how the seed
>> initialization works with randomu().

>

> Most interesting. What version of IDL are you using, Conor?

>

> Cheers,

>

> David

>

> --

> David Fanning, Ph.D.

> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

IDL Version 6.4 (linux x86 m32). (c) 2007, ITT Visual Information
Solutions

Subject: Re: Randomu() behavior - BAD!
Posted by [Jean H.](#) on Fri, 13 Nov 2009 20:11:30 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Conor writes:
>
>> So I did a simple test and have a better idea of how the seed
>> initialization works with randomu().
>
> Most interesting. What version of IDL are you using, Conor?
>
> Cheers,
>
> David

Same behavior here:

```
{ x86_64 Win32 Windows Microsoft Windows 7.0.8 Feb 9 2009    64    64}
```

Jean

Subject: Re: Randomu() behavior - BAD!
Posted by [David Fanning](#) on Fri, 13 Nov 2009 20:43:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Conor writes:

> So I did a simple test and have a better idea of how the seed
> initialization works with randomu(). In my mind, this is not good
> news!

I've written a new RandomNumberGenerator object this morning
that may get around some of these problems. (And avoid
common blocks, too!) You can find it here:

http://www.dfanning.com/programs/randomnumbergenerator_define.pro

There are probably any number of way you could use
this object, but here is one possibility. You could
create a new system variable that would be able to give
you random numbers on demand. The seed is always maintained
properly, so that the random number sequence will not repeat
in your lifetime. ;-)

```
IDL> DefSysV, !RNG, Obj_New('RandomNumberGenerator')  
IDL> Print, !RNG -> GetRandomNumbers(3)
```

0.089239137 0.77428782 0.069180504

You can set the initial seed, but if you don't, the number of seconds after 1 January 1970 is used. The random sequence will continue for as long as the object is alive in the IDL session.

You could create this object and system variable, for example, in your IDL startup script, and then any program you wrote could access it.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Randomu() behavior - BAD!

Posted by [Conor](#) on Fri, 13 Nov 2009 21:42:49 GMT

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

On Nov 13, 3:43 pm, David Fanning <n...@dfanning.com> wrote:

> Conor writes:

>> So I did a simple test and have a better idea of how the seed
>> initialization works with randomu(). In my mind, this is not good
>> news!

>

> I've written a new RandomNumberGenerator object this morning
> that may get around some of these problems. (And avoid
> common blocks, too!) You can find it here:

>

> http://www.dfanning.com/programs/randomnumbergenerator__define.pro

>

> There are probably any number of way you could use
> this object, but here is one possibility. You could
> create a new system variable that would be able to give
> you random numbers on demand. The seed is always maintained
> properly, so that the random number sequence will not repeat
> in your lifetime. ;-)

>

> IDL> DefSysV, !RNG, Obj_New('RandomNumberGenerator')

> IDL> Print, !RNG -> GetRandomNumbers(3)

> 0.089239137 0.77428782 0.069180504

>

> You can set the initial seed, but if you don't, the number
> of seconds after 1 January 1970 is used. The random sequence
> will continue for as long as the object is alive in the IDL
> session.
>
> You could create this object and system variable, for example,
> in your IDL startup script, and then any program you wrote
> could access it.
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:<http://www.dfanning.com/>
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Oooh, I like it!
