
Subject: Re: Random ordering

Posted by [David Fanning](#) on Tue, 24 Jun 2003 20:14:30 GMT

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

trouble writes:

> Stuck again. I want to create an integer 1D array from 0 to n (n ~
> 100) so that the numbers are in jumbled up order so that "nearby"
> numbers, eg. ABS(difference) < 3, are "far" apart. I think there must
> be a nice way using MOD rather than bludging it with RANDOMU. Anyone
> got any suggestions?

What is you do, anyway, for a living, Trouble?

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting, Inc.

Phone: 970-221-0438, E-mail: david@dfanning.com

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

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Random ordering

Posted by [James Kuyper](#) on Tue, 24 Jun 2003 20:55:36 GMT

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

David Fanning wrote:

>

> trouble writes:

>

>> Stuck again. I want to create an integer 1D array from 0 to n (n ~
>> 100) so that the numbers are in jumbled up order so that "nearby"
>> numbers, eg. ABS(difference) < 3, are "far" apart. I think there must
>> be a nice way using MOD rather than bludging it with RANDOMU. Anyone
>> got any suggestions?

Any algorithm than uses MOD rather than RANDOMU must necessarily be non-random, so I'm assuming a completely regular result is acceptable?

By "jumbled up" do you mean a permutation of indgen(n), as opposed to a sequence that might contain repeats?

If so, then try

```
jumble = (jumble_increment(n,m)*INDGEN(n)) MOD n
```

for m=3, which is the limit you've placed on ABS(difference).

`jumble_increment(n,m)` is the integer between 0 and n which is nearest to n/m , which does NOT divide 'n'. The algorithm for `jumble_increment()` is left as an excersize for the reader (I'm being lazy).
