
Subject: Re: Bizarre (?) behavior of randomu
Posted by [wita](#) on Fri, 04 Jan 2008 09:54:34 GMT
[View Forum Message](#) <> [Reply to Message](#)

The trick is indeed to specify a random seed during the first call to RANDOMU and preserve the seed value through subsequent calls to RANDOMU. If you do not specify it in the first call (like in your example) the seed value will be taken from some system values (like the systime or similar) and RANDOMU will produce different sets of random numbers with each call. This is generally undesirable, if you want your results to be reproducible.

example:

```
IDL> seed=1
```

```
IDL> print, randomu(seed, 10)
```

```
    0.415999  0.0919649  0.756410  0.529700  0.930436  
0.383502  0.653919  0.0668422  0.722660  0.671149
```

```
IDL> print, randomu(seed, 10)
```

```
    0.383416  0.631635  0.884707  0.519416  0.651519  
0.237774  0.262453  0.762198  0.753356  0.909208
```

```
IDL> seed=1
```

```
IDL> print, randomu(seed, 10)
```

```
    0.415999  0.0919649  0.756410  0.529700  0.930436  
0.383502  0.653919  0.0668422  0.722660  0.671149
```

```
IDL> print, randomu(seed, 10)
```

```
    0.383416  0.631635  0.884707  0.519416  0.651519  
0.237774  0.262453  0.762198  0.753356  0.909208
```

Note that while I specify seed=1, it will be replaced by a LONARR(36) after the first call to RANDOMU.

A more elegant solution without using COMMON blocks is to wrap the random number generator in an IDL object.

with best regards,

Allard
