

[3.14] Randomize list elements

Some simulations require a list of random integers, in which each integer appears only once. This program creates such a list:

```
randlist(n1,n2)
Prgm
©Create list 'm' of sequential integers in random order, from n1 to n2
©Idea by David Lloyd

local n

seq(k,k,n1,n2)→m
seq(rand(),k,1,dim(m))→n
sorta n,m

EndPrgm
```

$n1$ and $n2$ are the starting and ending values for the list elements. The resulting list is stored in the list m . For example, the call

```
randlist(0,4)
```

might return this for m :

```
{4, 2, 0, 1, 3}
```

The first `seq()` function creates a list of sequential integers from $n1$ to $n2$. The second `seq()` function creates list of random integers from 1 to the size of m . The `sorta` command sorts both lists, using the random list as the key, so that when the random list is sorted, the sequential list elements are randomized.

Ideally, this program would be written as a function to return the randomized list. Unfortunately, `sorta` is a command, not a function, and cannot be used in functions. The `sorta` arguments must be global variables, not local variables. 89/92+ functions cannot write to global variables, only local variables, so `sorta` will not work in a function.

None the less, this method is sufficiently short that it can be embedded into your programs. The necessary global variable can be deleted when your program exits.

(Credit to Daniel Lloyd)