RNORML	CERN Program Library	V120

Author(s) : F. James **Library:** MATHLIB
Submitter : **Submitted:** 15.03.1994
Language : Fortran **Revised:**

Gaussian-distributed Random Numbers

RNORML and RNORMX generate (vectors of) single-precision random numbers in a Gaussian distribution of mean zero and variance one. RNORML uses the uniform generator RANMAR underneath, and RNORMX allows the user to choose the uniform generator to be used underneath. The code is portable Fortran, but the results are not guaranteed to be identical on all platforms because there is branch on a floating-point compare which may (very rarely) cause the sequence produced on a given platform to be out of step with that of a different platform.

Structure:

SUBROUTINE Subprograms
User Entry Names: RNORML, RNORMX

Usage:

```
CALL RNORML(RVEC,LEN)
```

generates a vector RVEC of LEN Gaussian-distributed random numbers. RVEC is an array of type REAL and of length LEN at least.

The uniform generator used is RANMAR, so it may be initialized by calling RMARIN (V113), but beware that this also initializes RANMAR (V113)!

An alternative subroutine is supplied which allows the user to select the underlying uniform generator, for example RANLUX (V115).

```
EXTERNAL urng
...
CALL RNORMX(RVEC,LEN,urng)
```

where urng is a uniform random number generator of standard calling sequence: CALL urng(VEC,LENG). For example,

```
DIMENSION RVEC(10)
LEN = 10
EXTERNAL RANLUX
CALL RLUXGO(4,7675039,0,0)
  DO ...
     CALL RNORMX(RVEC,LEN,RANLUX)
```

would generate vectors of 10 Gaussian-distributed pseudorandom numbers of the highest quality. Note that initialization is now performed by the initializing entry for RANLUX, which is RLUXGO.

Method:

The method used to transform uniform deviates to Gaussian deviates is that known as the ratio of random deviates, discovered by Kinderman and Monahan, and improved by Leva (see **References**). The generation of one Gaussian random number requires at least two, and on average 2.74 uniform random numbers, as well as one floating-point division and on average 0.232 logarithm evaluations.

References:

1. J.L. Leva, A fast normal random number generator, ACM Trans. Math. Softw. **18** (1992) 449–453.
2. J.L. Leva, Algorithm 712. A normal random number generator, ACM Trans. Math. Softw. **18** (1992) 454–455.

•