3.3 Random Numbers: Exponential, Rayleigh, and Poisson

A. Purpose

Generate pseudorandom numbers from the exponential and Rayleigh distributions and pseudorandom integers from the Poisson distribution.

B. Usage

B.1 Generating exponential pseudorandom numbers

The density function for the exponential distribution with mean and standard deviation, μ, has the value zero for $x < 0$ and $\mu^{-1}\exp(-x/\mu)$ for $x \geq 0$. The cumulative distribution function has the value zero for $x < 0$ and $1 - \exp(-x/\mu)$ for $x \geq 0$. If u is a random variable having the uniform distribution on [0, 1] then $x = -\mu\ \log u$ is a random variable having the exponential distribution with mean and standard deviation, μ.

B.1.a Program Prototype, Single Precision

REAL SRANE, XMEAN, X

Assign a value to XMEAN.

X = SRANE(XMEAN)

B.1.b Argument Definitions

XMEAN [in] Specifies the mean and standard deviation of the desired exponential distribution. Require XMEAN > 0.

SRANE [out] The function returns a nonnegative pseudorandom number from the exponential distribution with mean and standard deviation equal to XMEAN.

B.2 Generating Rayleigh pseudorandom numbers

The density function for the Rayleigh distribution with scaling parameter, α, has the value zero for $x < 0$ and

$$(x/\alpha^2)\exp(-x^2/2\alpha^2)$$

for $x \geq 0$. The cumulative distribution function has the value zero for $x < 0$ and

$$1 - \exp(-x^2/2\alpha^2)$$

for $x \geq 0$. The mean and standard deviation of this distribution are

$$\mu = \alpha\sqrt{\pi/2} \approx 1.25331\ \alpha$$

and

$$\sigma = \alpha\sqrt{2-\pi/2} \approx 0.655136\ \alpha$$

If u is a random variable having the uniform distribution on [0, 1] then

$$x = \alpha\sqrt{-2\log u}$$

is a random variable having the Rayleigh distribution with scaling parameter, α.

B.2.a Program Prototype, Single Precision

REAL SRANR, ALPHA, X

Assign a value to ALPHA.

X = SRANR(ALPHA)

B.2.b Argument Definitions

ALPHA [in] Specifies the scaling of the desired Rayleigh distribution. Require ALPHA > 0. The distribution will have mean = ALPHA $\times\sqrt{\pi/2}$ and variance = ALPHA$^2 \times (2 - \pi/2)$.

SRANR [out] The function returns a nonnegative pseudorandom number from the Rayleigh distribution with scaling parameter, ALPHA.

B.3 Generating Poisson pseudorandom integers

The Poisson distribution with mean and variance, μ, is defined over nonnegative integers. The nonnegative integer, k, occurs with probability p_k given by

$$p_k = e^{-\mu}\frac{\mu^k}{k!}$$

B.3.a Program Prototype, Single Precision

REAL XMEAN

INTEGER ISRANP, K

Assign a value to XMEAN.

K = ISRANP(XMEAN)

B.3.b Argument Definitions

XMEAN [in] Specifies the mean and variance of the desired Poisson distribution. XMEAN must be positive and not so large that exp(−XMEAN) would underflow. For example if the underflow limit is 10^{-38}, XMEAN must not exceed 87. This subprogram requires more computing time for larger values of XMEAN or if XMEAN is changed frequently. See Section D.

ISRANP [out] The function returns a nonnegative pseudorandom integer from the Poisson distribution with mean XMEAN.

B.4 Modifications for Double Precision

Change the names SRANE, SRANR, and ISRANP to DRANE, DRANR, and IDRANP respectively, and change the REAL type statements above to DOUBLE PRECISION. Note particularly that if either of the function names DRANE or DRANR is used it must be typed DOUBLE PRECISION either explicitly or via an IMPLICIT statement.

C. Examples and Remarks

The programs DRSRANE, DRISRANP, and DRSRANR demonstrate, respectively, the use of SRANE, ISRANP, and SRANR. These programs use SSTAT1 and SSTAT2, or ISSTA1 and ISSTA2 to compute and print statistics and a histogram based on a sample of 10000 numbers each.

To fetch or set the seed used in the underlying pseudorandom integer sequence use the subroutines described in Chapter 3.1.

D. Functional Description

Method

The exponential random number is computed as $x = -\mu \log u$ where u is a random number from the uniform distribution on [0, 1].

The Rayleigh random number is computed as $x = \alpha\sqrt{-2\log u}$ where u is a random number from the uniform distribution on [0, 1].

The Poisson subprogram uses ideas from [1]. The method begins by obtaining a random number, u, from the uniform distribution on [0, 1]. Then the probabilities p_0, p_1, ..., defined above in Section B.3, are summed until the sum reaches or exceeds u. The index of the last term in the sum is then returned as the Poisson random integer.

To improve efficiency on the assumption that the subprogram may be referenced many successive times with the value of XMEAN remaining unchanged, the partial sums through at most the term p_{84} are stored in an internal array as they are computed. On subsequent references, if the value of XMEAN has not been changed, previously computed partial sums can be tested without the need to recompute them. The testing starts at the index nearest to the value, XMEAN, since these indices have the highest probabilities of being selected.

These subprograms obtain uniform pseudorandom numbers by calling SRANUA or DRANUA, using the array in common block /RANCMS/ or /RANCMD/ as a buffer as described in Chapter 3.1.

Values returned as double-precision random numbers will have random bits throughout the word, however the quality of randomness should not be expected to be as good in a low-order segment of the word as in a high-order part.

References

1. Richard H. Snow, *Algorithm 342 — Generator of random numbers satisfying the Poisson distribution*, **Comm. ACM 11**, 12 (Dec. 1968) 819.

E. Error Procedures and Restrictions

In subprograms SRANE, DRANE, SRANR, and DRANR the input parameter should be positive, however no test is made of this. The input parameter is simply used as a multiplicative factor.

Subprogram ISRANP will issue an error message and return the value −1 if XMEAN ≤ 0 or if XMEAN $\geq -0.5 \times \log($ underflow limit $)$.

F. Supporting Information

Entry	**Required Files**
DRANE	DRANE, ERFIN, ERMSG, RANPK1, RANPK2
DRANR	DRANR, ERFIN, ERMSG, RANPK1, RANPK2
IDRANP	AMACH, ERFIN, ERMSG, IDRANP, RANPK1, RANPK2, DERM1, DERV1
ISRANP	AMACH, ERFIN, ERMSG, ISRANP, RANPK1, RANPK2, SERM1, SERV1
SRANE	ERFIN, ERMSG, RANPK1, RANPK2, SRANE
SRANR	ERFIN, ERMSG, RANPK1, RANPK2, SRANR

Based on subprograms written for JPL by Stephen L. Richie, Heliodyne Corp., and Wiley R. Bunton, JPL, 1969. Adapted to Fortran 77 for the JPL MATH77 library by C. L. Lawson and S. Y. Chiu, JPL, April 1987.

1991 November: Lawson reorganized and renamed common blocks.

DRSRANE

```
c      program DRSRANE
c>> 2001-05-22 DRSRANE  Krogh Minor change for making .f90 version.
c>> 1996-05-28 DRSRANE  Krogh Added external statement.
c>> 1994-10-19 DRSRANE  Krogh  Changes to use M77CON
c>> 1987-12-09 DRSRANE  Lawson  Initial Code.
c--S replaces "?": DR?RANE, ?RANE, ?STAT1, ?STAT2
c
c      Driver to demonstrate use of SRANE to generate random numbers
c      from the exponential distribution with standard deviation, STDDEV.
c      Program computes histogram for N numbers
c      ------------------------------------------------------------------
      integer I, NCELLS
      parameter(NCELLS = 12+2)
      integer IHIST(NCELLS), N
      external SRANE
      real             SRANE, STATS(5), STDDEV, Y1, Y2, YTAB(1), ZERO
c
      parameter(ZERO = 0.0E0)
      data N / 10000 /
      data Y1, Y2 / 0.0E0, 6.0E0 /
      data STDDEV / 1.0E0 /
c      ------------------------------------------------------------------
      STATS(1) = ZERO
      do 20 I = 1, N
c                             Get random number
        YTAB(1) = SRANE(STDDEV)
c                          Accumulate statistics and histogram.
c
         call SSTAT1(YTAB(1), 1, STATS, IHIST, NCELLS, Y1, Y2)
   20 continue
c
      print '(11x,a)','Exponential random numbers from SRANE'
      print '(11x,a,g12.4,/1x)','with STDDEV = ',STDDEV
c
c                          Print the statistics and histogram.
c
      call SSTAT2(STATS, IHIST, NCELLS, Y1, Y2)
      stop
      end
```

ODSRANE

```
            Exponential random numbers from SRANE
            with STDDEV =    1.000

BREAK PT    COUNT            PLOT OF COUNT
    0.00 ---------------------------------------------------------------
                3930    0                                                  *
    0.50 ---------------------------------------------------------------
                2359    0                            *
    1.00 ---------------------------------------------------------------
                1465    0                  *
    1.50 ---------------------------------------------------------------
                 897    0           *
    2.00 ---------------------------------------------------------------
                 507    0      *
    2.50 ---------------------------------------------------------------
                 341    0    *
    3.00 ---------------------------------------------------------------
                 200    0*
    3.50 ---------------------------------------------------------------
                 116    0*
    4.00 ---------------------------------------------------------------
                  74    0*
    4.50 ---------------------------------------------------------------
                  51    *
    5.00 ---------------------------------------------------------------
                  27    *
    5.50 ---------------------------------------------------------------
                  14    *
    6.00 ---------------------------------------------------------------
                  19    *
```

Count	Minimum	Maximum	Mean	Std. Deviation
10000	0.97578E−04	8.1793	1.0001	0.98840

DRSRANR

```
c       program DRSRANR
c>> 2001-05-22 DRSRANR Krogh Minor change for making .f90 version.
c>> 1996-05-28 DRSRANR  Krogh Added external statement.
c>> 1994-10-19 DRSRANR  Krogh  Changes to use M77CON
c>> 1987-12-09 DRSRANR  Lawson  Initial Code.
c--S replaces "?": DR?RANR, ?RANR, ?STAT1, ?STAT2
c
c       Driver to demonstrate use of  SRANR to generate random numbers
c       from the Rayleigh distribution with parameter, ALPHA.
c       Program computes histogram for N numbers
c
      integer NCELLS
      parameter(NCELLS = 13+2)
      external SRANR
      real             ALPHA, SRANR, ONE, PIOV2
      real             STATS(5), TWO, Y1, Y2, YTAB(1), ZERO
      integer  I, IHIST(NCELLS), N
c
      parameter(ONE = 1.0E0, TWO = 2.0E0, ZERO = 0.0E0)
      data      N / 10000/
      data  Y1, Y2 /  0.0E0,   4.33333E0/
      data   ALPHA / 1.0E0 /
c     ------------------------------------------------------------------
      PIOV2 = TWO * atan(ONE)
      STATS(1) = ZERO
      do 20 I=1,N
c                                 Get random number
         YTAB(1) =  SRANR(ALPHA)
c
c                                 Accumulate statistics and histogram.
c
         call SSTAT1(YTAB(1), 1, STATS, IHIST, NCELLS, Y1, Y2)
   20 continue
c                                 Print the statistics and histogram.
c
      print '(13x,a)', 'Rayleigh random numbers from SRANR'
      print '(13x,a,g12.4)','with ALPHA = ',ALPHA
      print '(1x,a/13x,g13.5,a,g13.5/1x)',
     * 'The Mean and Std. Dev. of the theoretical distribution are',
     * ALPHA * sqrt(PIOV2), ' and ',ALPHA * sqrt(TWO - PIOV2)
      call SSTAT2(STATS, IHIST, NCELLS, Y1, Y2)
      stop
      end
```

ODSRANR

```
              Rayleigh random numbers from SRANR
              with ALPHA =      1.000
 The Mean and Std. Dev. of the theoretical distribution are
                1.2533      and    0.65514

 BREAK PT     COUNT          PLOT OF COUNT
   0.00 ------------------------------------------------------
               545     0        *
   0.33 ------------------------------------------------------
              1444     0                       *
   0.67 ------------------------------------------------------
              1941     0                               *
   1.00 ------------------------------------------------------
              1933     0                               *
   1.33 ------------------------------------------------------
              1614     0                         *
   1.67 ------------------------------------------------------
              1174     0                  *
   2.00 ------------------------------------------------------
               680     0          *
   2.33 ------------------------------------------------------
               382     0      *
   2.67 ------------------------------------------------------
               176     0   *
   3.00 ------------------------------------------------------
                81     0*
   3.33 ------------------------------------------------------
                25     *
   3.67 ------------------------------------------------------
                 4     *
   4.00 ------------------------------------------------------
                 1     *
   4.33 ------------------------------------------------------

  Count    Minimum       Maximum        Mean      Std. Deviation

  10000  0.13970E-01    4.0446        1.2537        0.65437
```

DRISRANP

```
c      program DRISRANP
c>> 1996-07-09 DRISRANP Krogh Added external statement.
c>> 1994-10-19 DRISRANP Krogh  Changes to use M77CON
c>> 1994-06-23 DRISRANP Changing name to DRI[D/S]RANP.
c>> 1987-12-09 DRISRANP  Lawson  Initial Code.
c
c      Driver to demonstrate use of ISRANP to generate random integers
c      from the Poisson distribution with mean, XMEAN.
c      Program computes histogram for N samples.
c      ------------------------------------------------------------------
c--S replaces "?": DRI?RANP, I?RANP, I?STA1, I?STA2
c      ------------------------------------------------------------------
      integer NCELLS, NI
      parameter(NCELLS = 7+2, NI = 100)
      external ISRANP
      integer I, IHIST(NCELLS), ISRANP, ISTATS(3), ITAB(NI), ILOW, J, N
      real              XMEAN, XSTATS(2)
c
      data N / 10000 /
      data ILOW / 0 /
      data XMEAN / 1.0e0 /
c      ------------------------------------------------------------------
      ISTATS(1) = 0
      do 30 J = 1, N/NI
         do 20 I = 1, NI
c                                     Get random integer.
            ITAB(I) =  ISRANP(XMEAN)
   20    continue
c                                  Accumulate statistics and histogram.
c
        call ISSTA1(ITAB, NI, ISTATS, XSTATS, IHIST, ILOW, NCELLS)
   30 continue
c                                  Print the statistics and histogram.
c
      print '(11x,''Poisson random integers from ISRANP'')'
      print '(11x,''with XMEAN = '',g12.4,/1x)', XMEAN
      call ISSTA2(ISTATS, XSTATS, IHIST, ILOW, NCELLS)
      end
```

ODISRANP

```
          Poisson random integers from ISRANP
          with XMEAN =      1.000

    VALUE      COUNT           PLOT OF COUNT
        0       3711     0                         *
        1       3635     0                        *
        2       1825     0             *
        3        634     0   *
        4        151     0*
        5         37     *
        6          7     *

      Count    Minimum   Maximum    Mean       Std. Deviation

      10000          0         6   1.0018          1.0099
```