1   SUMMARY

This function generates **uniformly distributed pseudo-random numbers.** Random numbers are generated in the ranges $0 < \xi < 1$, $-1 < \eta < 1$ and random integers in $1 \le k \le N$ where N is specified by the user.

A multiplicative congruent method is used where a 31 bit generator word g is maintained. On each call to the subroutine g_{n+1} is updated to $7^5 g_n \bmod(2^{31} - 1)$; the initial value of g is 1. Depending upon the type of random number required the following are computed $\xi = g_{n+1}/(2^{31} - 1)$; $\eta = 2\xi - 1$ or $k = \text{int.part}\{\xi N\} + 1$.

The subroutine also provides the facility for saving the current value of the generator word and for restarting with any specified value.

The subroutine is based on code found in L.Schrage, 'A More Portable Fortran Random Number Generator', TOMS, **5**, 2, June 1979.

ATTRIBUTES — **Version:** 1.0.1. (28 March 2013) **Types:** Real (single, double). **Original date:** March 2001. **Origin:** C.R.Kirby and C.L.Winskill, Harwell. **Licence:** A third-party licence for this package is available without charge. **Remark:** This is a threadsafe version of `FA04` and supersedes it.

2   HOW TO USE THE PACKAGE

2.1   Argument lists and calling sequences

There are five entries. The `FA14I`/`ID` entry must be called to initialize the generator word before any use is made of the other entries.

2.1.1   Initialize the generator word

The single precision version

```
CALL FA14I(SEED)
```

The double precision version

```
CALL FA14ID(SEED)
```

`SEED`   is an `INTEGER` variable which holds the generator word. It is considered to be private to `FA14` and should not be changed by the user.

2.1.2   Obtain random floating point numbers

The single precision version

```
R=FA14A(SEED,I)
```

The double precision version

```
R=FA14AD(SEED,I)
```

`SEED`   is an `INTEGER` variable which holds the generator word. It is considered to be private to `FA14` and should not be changed by the user.

`I`   is an `INTEGER` variable which must be set by the user to chose one of two ranges for the random number. Only its sign is significant. If `I` is non-negative the result is a real value in the range $0 < \xi < 1$ and if `I` is negative the result is a real value in the range $-1 < \eta < 1$.

`FA14A` is a `REAL` valued function subprogram returned set to the random number.

`FA14AD` is a `DOUBLE PRECISION` valued function subprogram returned set to the random number.

2.1.3 Obtain random integers

The single precision version

        CALL FA14B(SEED,N,K)

The double precision version

        CALL FA14BD(SEED,N,K)

`SEED` is an `INTEGER` variable which holds the generator word. It is considered to be private to `FA14` and should not be changed by the user.

`N`   is an `INTEGER` variable which must be set by the user to specify the upper limit N of the range of integers from which the random number is to be taken, i.e. the random number k is chosen from $1 \le k \le N$. **Restriction:** N must be positive.

`K`   is an `INTEGER` variable which will be set by the subroutine to the random integer k.

2.1.4 Save the current generator word

The single precision version

        CALL FA14C(SEED,IGEN)

The double precision version

        CALL FA14CD(SEED,IGEN)

`SEED` is an `INTEGER` variable which holds the generator word. It is considered to be private to `FA14` and should not be changed by the user.

`IGEN` is an `INTEGER` variable which will be set by the subroutine to the current value of the generator word g.

2.1.5 Reset the current value of the generator word

The single precision version

        CALL FA14D(SEED,IGEN)

The double precision version

        CALL FA14DD(SEED,IGEN)

`SEED` is an `INTEGER` variable which holds the generator word. It is considered to be private to `FA14` and should not be changed by the user.

`IGEN` is an `INTEGER` variable which must be set by the user to the new generator word. It must have a positive integer value less than $2^{31} - 1 \ (= 2147483647)$ and it is recommended that the value of `IGEN` should have been obtained by a previous call of `FA14C/CD`.

2.2 Maintaining the generator word

The initial value of the generator word must be set by calling the `FA14I/ID` entry before any use of the other entries. It is assumed that its value is not changed directly by the user but rather by calls to a combination of `FA14C/CD` and `FA14D/D`. In that case `SEED` should have a value of at least 1 and at most $2^{31} - 1 \ (=2147483646)$.

3 GENERAL INFORMATION

Use of common: none.

Workspace: none.

Other subprograms: none.

Input/Output: none.

Portability: The subroutines will work on any computer which is able to represent integers in the range $(-2^{31}, 2^{31})$. The Fortran code is nearly standard and should be portable to any computer which allows integers in the above range.

4 METHOD

4.1 Method description

The method employed is a multiplicative congruential method. The generator word g is held as an integer and is updated on each call to `FA14A` or `FA14B` as follows

$$g_{n+1} = 7^5 g_n \bmod (2^{31} - 1)$$

The result returned from `FA14A`, for a non-negative argument, is ξ, where

$$\xi = g_{n+1} / (2^{31} - 1)$$

and for a negative argument is

$$2\xi - 1$$

The value of k returned by `FA14B` is

$$\text{int.part}\{\xi N\} + 1$$

`FA14A/B/C/D/I` are based upon the Fortran coded random number generator published in T.O.M.S., **5**, no. 2, June 1979 (pages 132-138), by Linus Schrage.

4.2 Comparison with FA01A

`FA14A` provides the Fortran user with a random number generator which has a cycle length of $2^{31} - 1$ which is twice as long as the cycle length of `FA01A`.

5 EXAMPLE

This example program calculates the mean and variance of the first 1000 random numbers generated by FA14A.

```
C  TEST OF FA14A
      REAL S,T,X,XN
      INTEGER I,N,SEED
      REAL FA14A
      EXTERNAL FA14A
      CALL FA14I(SEED)
      S = 0.0
      T = 0.0
      N = 1000
      XN = N
      DO 50 I = 1,N
        X = FA14A(SEED,I)
        T = T + (X-0.5)**2
        S = S + X
   50 CONTINUE
      S = S/XN
      T = T/XN
      WRITE (6,FMT=1) N,S,T
    1 FORMAT (5X,'MEAN OF',I6,' NUMBERS=',E16.7,5X,'VAR=',E16.7)
      END
```

This produces the following output:

```
    MEAN OF  1000 NUMBERS=   0.4979618E+00    VAR=   0.7875296E-01
```