NAG C Library Function Document

nag_rns_discrete_uniform (g05mac)

1 Purpose

nag_rns_discrete_uniform (g05mac) generates a vector of pseudo-random integers uniformly distributed over the interval \([a, b]\).

2 Specification

```c
void nag_rns_discrete_uniform (Integer a, Integer b, Integer n, Integer x[],
       Integer igen, Integer iseed[], NagError *fail)
```

3 Description

nag_rns_discrete_uniform (g05mac) generates the next \(n\) values \(y_i\) from a uniform \((0,1)\) generator (see nag_rns_basic (g05kac) for details) and applies the transformation

\[ x_i = a + \lfloor (b - a + 1) y_i \rfloor, \]

where \(\lfloor z \rfloor\) is the integer part of the real value \(z\). The function ensures that the values \(x_i\) lie in the closed interval \([a, b]\).

One of the initialisation functions nag_rns_init_repeatable (g05kbc) (for a repeatable sequence if computed sequentially) or nag_rns_init_nonrepeatable (g05kcc) (for a non-repeatable sequence) must be called prior to the first call to nag_rns_discrete_uniform (g05mac).

4 References


5 Parameters

1:  \(a\) – Integer  
    \(\text{Input}\)

   On entry: the end-points \(a\) and \(b\) of the uniform distribution.

   Constraint: \(a \leq b\).

2:  \(b\) – Integer  
    \(\text{Input}\)

   On entry: the end-points \(a\) and \(b\) of the uniform distribution.

   Constraint: \(a \leq b\).

3:  \(n\) – Integer  
    \(\text{Input}\)

   On entry: the number, \(n\), of pseudo-random numbers to be generated.

   Constraint: \(n \geq 0\).

4:  \(x[\text{dim}]\) – Integer  
    \(\text{Output}\)

   Note: the dimension, \(\text{dim}\), of the array \(x\) must be at least \(\max(1, n)\).

   On exit: the \(n\) pseudo-random numbers from the specified uniform distribution.

5:  \(igen\) – Integer  
    \(\text{Input}\)

   On entry: must contain the identification number for the generator to be used to return a pseudo-random number and should remain unchanged following initialisation by a prior call to one of the functions nag_rns_init_repeatable (g05kbc) or nag_rns_init_nonrepeatable (g05kcc).

6:  \(iseed[4]\) – Integer  
    \(\text{Input/Output}\)

   On entry: contains values which define the current state of the selected generator.
On exit: contains updated values defining the new state of the selected generator.

7: fail – NagError *

The NAG error parameter (see the Essential Introduction).

6 Error Indicators and Warnings

NE_INT

On entry, \( n = \langle\text{value}\rangle \).
Constraint: \( n \geq 0 \).

NE_INT_2

On entry, \( a = \langle\text{value}\rangle \) and \( b = \langle\text{value}\rangle \).
Constraint: \( b \geq a \).

NE_BAD_PARAM

On entry, parameter \( \langle\text{value}\rangle \) had an illegal value.

NE_INTERNAL_ERROR

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please consult NAG for assistance.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

The example program prints five pseudo-random integers from a discrete uniform distribution between \(-5\) and \(5\), generated by a single call to nag_rngs_discrete_uniform (g05mac), after initialisation by nag_rngs_init_repeatable (g05kbc).

9.1 Program Text

```c
#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nagg05.h>

int main(void)
{
    /* Scalars */
    Integer igen, j, m;
    Integer exit_status=0;
    NagError fail;
```
g05mac

Integer  *x=0;
Integer  iseed[4];

INIT_FAIL(fail);
Vprintf("g05mac Example Program Results\n\n");

m = 5;
/* Allocate memory */
if ( !(x = NAG_ALLOC(m, Integer)) )
{  
  Vprintf("Allocation failure\n");
  exit_status = -1;
  goto END;
}

/* Initialise the seed to a repeatable sequence */
iseed[0] = 1762543;
iseed[1] = 9324783;
iseed[2] = 42344;
iseed[3] = 742355;
/* igen identifies the stream. */
igen = 1;
g05kbc(&igen, iseed);
g05mac(-5, 5, m, x, igen, iseed, &fail);
if (fail.code != NE_NOERROR)
{  
  Vprintf("Error from g05mac.\n\n", fail.message);
  exit_status = 1;
  goto END;
}
for (j = 0; j < m; ++j)
{  
  Vprintf("%12ld\n", x[j]);
}

END:
if (x) NAG_FREE(x);
return exit_status;

9.2 Program Data
None.

9.3 Program Results

g05mac Example Program Results

-5
  5
-1
  3
  5