Utility: nagdmc_rank_real

Purpose

nagdmc_rank_real computes the ranks for an array of real values.

Declaration

```
#include <nagdmc.h>
void nagdmc_rank_real(long n, long start, long step, double data[], long rank[],
                      int *info);
```

Parameters

1: **n** – `long` *Input*
On entry: the number of data values to arrange in rank order.
Constraint: **n** > 0.

2: **start** – `long` *Input*
On entry: **data**[**start**] is the first data value considered.
Constraint: **start** ≥ 0.

3: **step** – `long` *Input*
On entry: the step length between pairs of data values.
Constraint: **step** ≥ 1.

4: **data**[d] – `double` *Input*
On entry: an array of length $d \geq (\mathbf{n}-1) * \mathbf{step} + \mathbf{start} + 1$ containing real values.

5: **rank**[**n**] – `long` *Output*
On exit: the ranks for the **n** considered data values in **data**.

6: **info** – `int *` *Output*
On exit: **info** gives information on the success of the function call:

0: the function successfully completed its task.
i; $i = 1, 2, 3$: the specification of the ith formal parameter was incorrect.

Notation

n the number of data values, n.
data data values x_i (separated by **step** values) to rank, for $i = 1, 2, \ldots, n$.
rank rank values r_i, for $i = 1, 2, \ldots, n$.

Description

Given n data values x_i, for $i = 1, 2, \ldots, n$, we seek a permutation,

$$p(1), p(2), \ldots, p(n)$$

that arranges the data values in ascending order, i.e.,

$$x_{p(1)} \leq x_{p(2)} \leq \ldots \leq x_{p(n)}.$$

The position of the ith data value x_i in the permuted order is its rank value, r_i, for $i = 1, 2, \ldots, n$.

References and Further Reading

Knuth D E (1973) *The Art of Computer Programming (Volume 3)* (2nd Edition) Addison-Wesley.

See Also

nagdmc_index converts a ranks into indexes in sorted order.
nagdmc_order_real re-arranges an array of data values according to rank values.