NAG Library Function Document

nag_enum_value_to_name (x04nbc)

1 Purpose

nag_enum_value_to_name (x04nbc) returns the name of the NAG enumeration member given the enumeration member's value.

2 Specification

```
#include <nag.h>
#include <nagx04.h>
```

```
const char * nag_enum_value_to_name (int enum_value)
```

3 Description

nag_enum_value_to_name (x04nbc) takes an integer argument, which must be the value of a NAG enumeration member and returns a string which is the name of that member (e.g., `"Nag_ColMajor"`, `"Nag_LogNormal"`, etc.). If the input value does not correspond to a NAG enumeration member then the function returns 0.

The reverse process of converting from enumeration member name to enumeration member value is also available using nag_enum_name_to_value (x04nac).

Converting enumeration members to and from name and value may be of use when saving a set of problem arguments to file or reading problem arguments from a file for use in an application. In the case of saving problem arguments, any enumeration members to be saved should be saved using their names to be subsequently read as strings. nag_enum_value_to_name (x04nbc) can be used to get enumeration member names for writing.

4 References

None.

5 Arguments

1: **enum_value** – int *Input*

On entry: the value of a NAG enumeration member.

6 Error Indicators and Warnings

If the value 0 is returned then the input value is not recognized as a valid NAG enumeration member value.

7 Accuracy

Not applicable.

8 Parallelism and Performance

Not applicable.

9 Further Comments

None.

10 Example

This example takes a set of NAG enumeration members and checks that the value used has a name that matches the member name.

10.1 Program Text

```
/* nag_enum_value_to_name (x04nbc) Example Program.
 *
 * Copyright 2014 Numerical Algorithms Group.
 *
 * Mark 8, 2005.
 */

#include <stdio.h>
#include <nag.h>
#include <nag_stdlib.h>
#include <nag_string.h>
#include <nagx04.h>

int main(void)
{
  /* Scalars */
  Integer        exit_status = 0;
  Nag_OrderType  order;
  Nag_TransType  trans;
  Nag_DiagType   unitdiag;
  Nag_MatrixType matrix;

  /* Pointers */
  const char     *str_order, *str_trans, *str_matrix, *str_unitdiag;
  const char     *status;

  printf("nag_enum_value_to_name (x04nbc) Example Program Results\n\n");

  /* Set some Nag types using enum member names */
  order = Nag_ColMajor;
  trans = Nag_Trans;
  matrix = Nag_GeneralMatrix;
  unitdiag = Nag_NonUnitDiag;

  /* Convert the values held by these typed variables to strings */

  /* nag_enum_value_to_name (x04nbc).
   * Converts NAG enum member value to its name
   */
  str_order = nag_enum_value_to_name(order);
  str_trans = nag_enum_value_to_name(trans);
  str_matrix = nag_enum_value_to_name(matrix);
  str_unitdiag = nag_enum_value_to_name(unitdiag);

  /* Check strings match member names and print. */
  printf("   Member name     String returned  Status\n");
  printf("----------------- ----------------- ------\n");
  status = "OK";
  if (strcmp(str_order, "Nag_ColMajor"))
    {
      status = "Error";
      exit_status++;
    }
  printf("%-17s %-17s %-6s\n", "Nag_ColMajor", str_order, status);

  status = "OK";
  if (strcmp(str_trans, "Nag_Trans"))
```

```
    {
      status = "Error";
      exit_status++;
    }
  printf("%-17s %-17s %-6s\n", "Nag_Trans", str_trans, status);

  status = "OK";
  if (strcmp(str_matrix, "Nag_GeneralMatrix"))
    {
      status = "Error";
      exit_status++;
    }
  printf("%-17s %-17s %-6s\n", "Nag_GeneralMatrix", str_matrix, status);

  status = "OK";
  if (strcmp(str_unitdiag, "Nag_NonUnitDiag"))
    {
      status = "Error";
      exit_status++;
    }
  printf("%-17s %-17s %-6s\n", "Nag_NonUnitDiag", str_unitdiag, status);

  return exit_status;
}
```

10.2 Program Data

None.

10.3 Program Results

```
nag_enum_value_to_name (x04nbc) Example Program Results

   Member name     String returned  Status
----------------- ----------------- ------
Nag_ColMajor      Nag_ColMajor      OK
Nag_Trans         Nag_Trans         OK
Nag_GeneralMatrix Nag_GeneralMatrix OK
Nag_NonUnitDiag   Nag_NonUnitDiag   OK
```