

NAG Fortran Library Routine Document

X05ABF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

1 Purpose

X05ABF converts from a seven-integer format time and date, as returned by X05AAF, into a character string, returned via the routine name.

2 Specification

```
CHARACTER*30 FUNCTION X05ABF(ITIME)
INTEGER                               ITIME(7)
```

3 Description

X05ABF returns a character string of length 30 which contains the date and time as supplied in argument ITIME. On exit, the character string has the following format:

```
'DAY XXTH MTH YEAR HR:MN:SC.MIL'
```

where

DAY is one of 'Sun', 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat',

XX is an integer denoting the day of the month,

TH is one of 'st', 'nd', 'rd', 'th',

MTH is one of 'Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec',

YEAR is the year as a four digit integer,

HR is the hour,

MN is the minute,

SC is the second,

MIL is the millisecond.

If on entry the date in ITIME is invalid, the string returned is '** Illegal date/time **'

4 References

None.

5 Parameters

1: ITIME(7) – INTEGER array

Input

On entry: a date and time in the format returned by X05AAF, as follows:

ITIME(1) must contain the year as a positive integer.

ITIME(2) must contain the month, in the range 1–12.

ITIME(3) must contain the day, in the range 1 to p , where $p = 28, 29, 30$ or 31 , depending on the month and year.

ITIME(4) must contain the hour, in the range 0–23.

ITIME(5) must contain the minute, in the range 0–59.

ITIME(6) must contain the second, in the range 0–59.

ITIME(7) must contain the millisecond, in the range 0–999.

6 Error Indicators and Warnings

None.

7 Accuracy

The day name included as part of the character string returned by this routine is calculated assuming that the date is part of the Gregorian calendar. This calendar has been in operation in Europe since 15 October 1582, and in Great Britain since 14 September 1752. Entry to this routine with a date earlier than these will therefore not return a day name that is historically accurate.

8 Further Comments

Two dates stored in character string format, as returned by this routine, may be compared by X05ACF.

9 Example

The example program initialises a time in ITIME, and converts it to character format by a call to X05ABF.

9.1 Program Text

Note: the listing of the example program presented below uses *bold italicised* terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*      X05ABF Example Program Text
*      Mark 14 Release.  NAG Copyright 1989.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER       (NIN=5,NOUT=6)
*      .. Local Scalars ..
      CHARACTER*30    CTIME
*      .. Local Arrays ..
      INTEGER          ITIME(7)
*      .. External Functions ..
      CHARACTER*30    X05ABF
      EXTERNAL        X05ABF
*      .. Executable Statements ..
      WRITE (NOUT,*) 'X05ABF Example Program Results'
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) ITIME
*
      CTIME = X05ABF(ITIME)
*
      WRITE (NOUT,99999) CTIME
      STOP
*
99999  FORMAT (1X,A)
      END
```

9.2 Program Data

```
X05ABF Example Program Data
1789 7 14 13 11 48 320
```

9.3 Program Results

X05ABF Example Program Results
Tue 14th Jul 1789 13:11:48.320
