NAG Fortran Library Routine Document

X05ACF

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

X05ACF compares two date/time character strings, each stored in the format returned by X05ABF.

2 Specification

```
INTEGER FUNCTION X05ACF(CTIME1, CTIME2)
CHARACTER*(*)           CTIME1, CTIME2
```

3 Description

X05ACF compares two date/time character strings, and returns an integer that specifies which one is the earliest. The result is an integer returned through the routine name, with meaning as follows:

if X05ACF $= -1$, the first date/time string is earlier than the second;

if X05ACF $= 0$, the two date/time strings are equivalent;

if X05ACF $= 1$, the first date/time string is later than the second.

4 References

None.

5 Parameters

1: CTIME1 – CHARACTER*(*) *Input*
2: CTIME2 – CHARACTER*(*) *Input*

On entry: the date/time strings to be compared. These are expected be in the format returned by X05ABF, although X05ACF will still attempt to interpret the strings if they vary slightly from this format. See Section 8 for further details.

6 Error Indicators and Warnings

None.

7 Accuracy

Not applicable.

8 Further Comments

For flexibility, X05ACF will accept various formats for the two date/time strings CTIME1 and CTIME2.

The strings do not have to be the same length. It is permissible, for example, to enter with one or both of the strings truncated to a smaller length, in which case missing fields are treated as zero.

Each character string may be of any length, but everything after character 80 is ignored.

Each string may or may not include an alphabetic day name, such as 'Wednesday', at its start. These day names are ignored, and no check is made that the day name corresponds correctly to the rest of the date.

The month name may contain any number of characters provided it uniquely identifies the month, however all characters that are supplied are significant.

Fields in the character string must be separated by one or more spaces.

The case of all alphabetic characters is not significant.

Any field in a date time string that is indecipherable according to the above rules will be converted to a zero value internally. Thus two strings that are completely indecipherable will compare equal.

According to these rules, all the following date/time strings are equivalent:

'Thursday 10th July 1958 12:43:17.320'

'THU 10th JULY 1958 12:43:17.320'

'10th Jul 1958 12:43:17.320'

9 Example

The example program initialises two date/time strings, and compares them by a call to X05ACF.

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.

```
*     X05ACF Example Program Text
*     Mark 14 Release.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      INTEGER          K
      CHARACTER*50     CTIME1, CTIME2
*     .. External Functions ..
      INTEGER          X05ACF
      EXTERNAL         X05ACF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'X05ACF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) CTIME1, CTIME2
*
      K = X05ACF(CTIME1,CTIME2)
*
      IF (K.LT.0) THEN
         WRITE (NOUT,99999) CTIME1
         WRITE (NOUT,99999) 'is earlier than'
         WRITE (NOUT,99999) CTIME2
      ELSE IF (K.EQ.0) THEN
         WRITE (NOUT,99999) CTIME1
         WRITE (NOUT,99999) 'is equivalent to'
         WRITE (NOUT,99999) CTIME2
      ELSE
         WRITE (NOUT,99999) CTIME1
         WRITE (NOUT,99999) 'is later than'
         WRITE (NOUT,99999) CTIME2
      END IF
      STOP
*
99999 FORMAT (1X,A)
      END
```

9.2 Program Data

```
X05ACF Example Program Data
'Thu 27th April 1989 13:15:21.320'
'Wed 26th April 1989 11:23:14.130'
```

9.3 Program Results

```
 X05ACF Example Program Results
 Thu 27th April 1989 13:15:21.320
 is later than
 Wed 26th April 1989 11:23:14.130
```