NAG Fortran Library Routine Document

F08AJF (SORGLQ/DORGLQ)

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

F08AJF (SORGLQ/DORGLQ) generates all or part of the real orthogonal matrix Q from an LQ factorization computed by F08AHF (SGELQF/DGELQF).

2 Specification

```
SUBROUTINE F08AJF(M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
ENTRY      sorglq (M, N, K, A, LDA, TAU, WORK, LWORK, INFO)
INTEGER           M, N, K, LDA, LWORK, INFO
real              A(LDA,*), TAU(*), WORK(*)
```

The ENTRY statement enables the routine to be called by its LAPACK name.

3 Description

This routine is intended to be used after a call to F08AHF (SGELQF/DGELQF), which performs an LQ factorization of a real matrix A. The orthogonal matrix Q is represented as a product of elementary reflectors.

This routine may be used to generate Q explicitly as a square matrix, or to form only its leading rows.

Usually Q is determined from the LQ factorization of a p by n matrix A with $p \leq n$. The whole of Q may be computed by:

```
CALL SORGLQ (N,N,P,A,LDA,TAU,WORK,LWORK,INFO)
```

(note that the array A must have at least n rows) or its leading p rows by:

```
CALL SORGLQ (P,N,P,A,LDA,TAU,WORK,LWORK,INFO)
```

The rows of Q returned by the last call form an orthonormal basis for the space spanned by the rows of A; thus F08AHF (SGELQF/DGELQF) followed by F08AJF (SORGLQ/DORGLQ) can be used to orthogonalise the rows of A.

The information returned by the LQ factorization routines also yields the LQ factorization of the leading k rows of A, where $k < p$. The orthogonal matrix arising from this factorization can be computed by:

```
CALL SORGLQ (N,N,K,A,LDA,TAU,WORK,LWORK,INFO)
```

or its leading k rows by:

```
CALL SORGLQ (K,N,K,A,LDA,TAU,WORK,LWORK,INFO)
```

4 References

Golub G H and van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

1: M – INTEGER *Input*

On entry: m, the number of rows of the matrix Q.

Constraint: $M \geq 0$.

2: N – INTEGER *Input*

On entry: n, the number of columns of the matrix Q.

Constraint: $N \geq M$.

3: K – INTEGER *Input*

On entry: k, the number of elementary reflectors whose product defines the matrix Q.

Constraint: $M \geq K \geq 0$.

4: A(LDA,*) – ***real*** array *Input/Output*

Note: the second dimension of the array A must be at least $\max(1, N)$.

On entry: details of the vectors which define the elementary reflectors, as returned by F08AHF (SGELQF/DGELQF).

On exit: the m by n matrix Q.

5: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08AJF (SORGLQ/DORGLQ) is called.

Constraint: $LDA \geq \max(1, M)$.

6: TAU(*) – ***real*** array *Input*

Note: the dimension of the array TAU must be at least $\max(1, K)$.

On entry: further details of the elementary reflectors, as returned by F08AHF (SGELQF/DGELQF).

7: WORK(*) – ***real*** array *Workspace*

Note: the dimension of the array WORK must be at least $\max(1, LWORK)$.

On exit: if INFO = 0, WORK(1) contains the minimum value of LWORK required for optimum performance.

8: LWORK – INTEGER *Input*

On entry: the dimension of the array WORK as declared in the (sub)program from which F08AJF (SORGLQ/DORGLQ) is called, unless LWORK $= -1$, in which case a workspace query is assumed and the routine only calculates the optimal dimension of WORK (using the formula given below).

Suggested value: for optimum performance LWORK should be at least $M \times nb$, where nb is the ***blocksize***.

Constraint: $LWORK \geq \max(1, M)$ or $LWORK = -1$.

9: INFO – INTEGER *Output*

On exit: INFO = 0 unless the routine detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If INFO $= -i$, the ith parameter had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

The computed matrix Q differs from an exactly orthogonal matrix by a matrix E such that

$$\|E\|_2 = O(\epsilon),$$

where ϵ is the ***machine precision***.

8 Further Comments

The total number of floating-point operations is approximately $4mnk - 2(m+n)k^2 + \frac{4}{3}k^3$; when $m = k$, the number is approximately $\frac{2}{3}m^2(3n - m)$.

The complex analogue of this routine is F08AWF (CUNGLQ/ZUNGLQ).

9 Example

To form the leading 4 rows of the orthogonal matrix Q from the LQ factorization of the matrix A, where

$$A = \begin{pmatrix} -5.42 & 3.28 & -3.68 & 0.27 & 2.06 & 0.46 \\ -1.65 & -3.40 & -3.20 & -1.03 & -4.06 & -0.01 \\ -0.37 & 2.35 & 1.90 & 4.31 & -1.76 & 1.13 \\ -3.15 & -0.11 & 1.99 & -2.70 & 0.26 & 4.50 \end{pmatrix}.$$

The rows of Q form an orthonormal basis for the space spanned by the rows of A.

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.

```
*     F08AJF Example Program Text
*     Mark 16 Release. NAG Copyright 1992.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX, LDA, LWORK
      PARAMETER        (MMAX=8,NMAX=8,LDA=MMAX,LWORK=64*MMAX)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, INFO, J, M, N
      CHARACTER*30     TITLE
*     .. Local Arrays ..
      real             A(LDA,NMAX), TAU(NMAX), WORK(LWORK)
*     .. External Subroutines ..
      EXTERNAL         sgelqf, sorglq, X04CAF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'F08AJF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. M.LE.N) THEN
*
*        Read A from data file
*
         READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
*
```

```
*        Compute the LQ factorization of A
*
         CALL sgelqf(M,N,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Form the leading M rows of Q explicitly
*
         CALL sorglq(M,N,M,A,LDA,TAU,WORK,LWORK,INFO)
*
*        Print the leading M rows of Q only
*
         WRITE (NOUT,*)
         WRITE (TITLE,99999) M
         IFAIL = 0
*
         CALL X04CAF('General',' ',M,N,A,LDA,TITLE,IFAIL)
*
      END IF
      STOP
*
99999 FORMAT ('The leading ',I2,' rows of Q')
      END
```

9.2 Program Data

```
F08AJF Example Program Data
  4  6                                                 :Values of M and N
 -5.42   3.28  -3.68   0.27   2.06   0.46
 -1.65  -3.40  -3.20  -1.03  -4.06  -0.01
 -0.37   2.35   1.90   4.31  -1.76   1.13
 -3.15  -0.11   1.99  -2.70   0.26   4.50   :End of matrix A
```

9.3 Program Results

```
 F08AJF Example Program Results

 The leading  4 rows of Q
          1       2       3       4       5       6
 1  -0.7104  0.4299 -0.4824  0.0354  0.2700  0.0603
 2  -0.2412 -0.5323 -0.4845 -0.1595 -0.6311 -0.0027
 3   0.1287 -0.2619 -0.2108 -0.7447  0.5227 -0.2063
 4  -0.3403 -0.0921  0.4546 -0.3869 -0.0465  0.7191
```