

NAG Library Routine Document

F08YSF (ZTGSJA)

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

F08YSF (ZTGSJA) computes the generalized singular value decomposition (GSVD) of two complex upper trapezoidal matrices A and B , where A is an m by n matrix and B is a p by n matrix.

A and B are assumed to be in the form returned by F08VSF (ZGGSVP).

2 Specification

```

SUBROUTINE F08YSF(JOBU, JOBV, JOBQ, M, P, N, K, L, A, LDA, B, LDB, TOLA,
1              TOLB, ALPHA, BETA, U, LDU, V, LDV, Q, LDQ, WORK,
2              NCYCLE, INFO)
    INTEGER      M, P, N, K, L, LDA, LDB, LDU, LDV, LDQ, NCYCLE, INFO
    double precision TOLA, TOLB, ALPHA(N), BETA(N)
    complex*16      A(LDA,*), B(LDB,*), U(LDU,*), V(LDV,*), Q(LDQ,*),
1              WORK(2*N)
    CHARACTER*1  JOBU, JOBV, JOBQ

```

The routine may be called by its LAPACK name *ztgsja*.

3 Description

F08YSF (ZTGSJA) computes the GSVD of the matrices A and B which are assumed to have the form as returned by F08VSF (ZGGSVP)

$$A = \begin{cases} \begin{pmatrix} n-k-l & k & l \\ k & 0 & A_{12} & A_{13} \\ l & 0 & 0 & A_{23} \\ m-k-l & 0 & 0 & 0 \end{pmatrix}, & \text{if } m-k-l \geq 0; \\ \begin{pmatrix} n-k-l & k & l \\ k & 0 & A_{12} & A_{13} \\ m-k & 0 & 0 & A_{23} \end{pmatrix}, & \text{if } m-k-l < 0; \end{cases}$$

$$B = \begin{pmatrix} n-k-l & k & l \\ l & 0 & 0 & B_{13} \\ p-l & 0 & 0 & 0 \end{pmatrix},$$

where the k by k matrix A_{12} and the l by l matrix B_{13} are nonsingular upper triangular, A_{23} is l by l upper triangular if $m-k-l \geq 0$ and is $(m-k)$ by l upper trapezoidal otherwise.

F08YSF (ZTGSJA) computes unitary matrices Q , U and V , diagonal matrices D_1 and D_2 , and an upper triangular matrix R such that

$$U^H A Q = D_1 \begin{pmatrix} 0 & R \end{pmatrix}, \quad V^H B Q = D_2 \begin{pmatrix} 0 & R \end{pmatrix}.$$

Optionally Q , U and V may or may not be computed, or they may be premultiplied by matrices Q_1 , U_1 and V_1 respectively.

If $(m - k - l) \geq 0$ then D_1 , D_2 and R have the form

$$D_1 = \begin{matrix} & k & l \\ & \begin{pmatrix} I & 0 \\ 0 & C \end{pmatrix} \\ m - k - l & \begin{pmatrix} 0 & 0 \end{pmatrix} \end{matrix},$$

$$D_2 = \begin{matrix} & k & l \\ l & \begin{pmatrix} 0 & S \\ 0 & 0 \end{pmatrix} \\ p - l & \begin{pmatrix} 0 & 0 \end{pmatrix} \end{matrix},$$

$$R = \begin{matrix} & k & l \\ k & \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix} \\ l & \begin{pmatrix} 0 & R_{22} \end{pmatrix} \end{matrix},$$

where $C = \text{diag}(\alpha_{k+1}, \dots, \alpha_{k+l})$, $S = \text{diag}(\beta_{k+1}, \dots, \beta_{k+l})$.

If $(m - k - l) < 0$ then D_1 , D_2 and R have the form

$$D_1 = \begin{matrix} & k & m - k & k + l - m \\ & \begin{pmatrix} I & 0 & 0 \\ 0 & C & 0 \end{pmatrix} \\ m - k & \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \end{matrix},$$

$$D_2 = \begin{matrix} & k & m - k & k + l - m \\ m - k & \begin{pmatrix} 0 & S & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{pmatrix} \\ k + l - m & \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \\ p - l & \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \end{matrix},$$

$$R = \begin{matrix} & k & m - k & k + l - m \\ & \begin{pmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{pmatrix} \\ m - k & \begin{pmatrix} 0 & R_{22} & R_{23} \end{pmatrix} \\ k + l - m & \begin{pmatrix} 0 & 0 & R_{33} \end{pmatrix} \end{matrix},$$

where $C = \text{diag}(\alpha_{k+1}, \dots, \alpha_m)$, $S = \text{diag}(\beta_{k+1}, \dots, \beta_m)$.

In both cases the diagonal matrix C has real non-negative diagonal elements, the diagonal matrix S has real positive diagonal elements, so that S is nonsingular, and $C^2 + S^2 = 1$. See Section 2.3.5.3 of Anderson *et al.* (1999) for further information.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D (1999) *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia <http://www.netlib.org/lapack/lug>

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

5 Parameters

1: JOBU – CHARACTER*1

Input

On entry: if JOBU = 'U', U must contain a unitary matrix U_1 on entry, and the product $U_1 U$ is returned.

If JOBU = 'I', U is initialized to the unit matrix, and the unitary matrix U is returned.

If $\text{JOBV} = \text{'N'}$, U is not computed.

Constraint: $\text{JOBV} = \text{'I'}$, 'U' or 'N' .

2: $\text{JOBV} - \text{CHARACTER*1}$

Input

On entry: if $\text{JOBV} = \text{'V'}$, V must contain a unitary matrix V_1 on entry, and the product $V_1 V$ is returned.

If $\text{JOBV} = \text{'I'}$, V is initialized to the unit matrix, and the unitary matrix V is returned.

If $\text{JOBV} = \text{'N'}$, V is not computed.

Constraint: $\text{JOBV} = \text{'V'}$ or 'N' .

3: $\text{JOBQ} - \text{CHARACTER*1}$

Input

On entry: if $\text{JOBQ} = \text{'Q'}$, Q must contain a unitary matrix Q_1 on entry, and the product $Q_1 Q$ is returned.

If $\text{JOBQ} = \text{'I'}$, Q is initialized to the unit matrix, and the unitary matrix Q is returned.

If $\text{JOBQ} = \text{'N'}$, Q is not computed.

Constraint: $\text{JOBQ} = \text{'Q'}$ or 'N' .

4: $M - \text{INTEGER}$

Input

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

Constraint: $M \geq 0$.

5: $P - \text{INTEGER}$

Input

On entry: p , the number of rows of the matrix B .

Constraint: $P \geq 0$.

6: $N - \text{INTEGER}$

Input

On entry: n , the number of columns of the matrices A and B .

Constraint: $N \geq 0$.

7: $K - \text{INTEGER}$

Input

8: $L - \text{INTEGER}$

Input

On entry: K and L specify the sizes, k and l , of the subblocks of A and B , whose GSVD is to be computed by F08YSF (ZTGSJA).

9: $A(\text{LDA},*) - \text{complex*16 array}$

Input/Output

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

On entry: the m by n matrix A .

On exit: if $m - k - l \geq 0$, $A(1 : k + l, n - k - l + 1 : n)$ contains the $(k + l)$ by $(k + l)$ upper triangular matrix R .

If $m - k - l < 0$, $A(1 : m, n - k - l + 1 : n)$ contains the first m rows of the $(k + l)$ by $(k + l)$ upper triangular matrix R , and the submatrix R_{33} is returned in $B(m - k + 1 : l, n + m - k - l + 1 : n)$.

10: $\text{LDA} - \text{INTEGER}$

Input

On entry: the first dimension of the array A as declared in the (sub)program from which F08YSF (ZTGSJA) is called.

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

- 11: B(LDB,*) – **complex*16** array Input/Output
Note: the second dimension of the array B must be at least $\max(1, N)$.
On entry: the p by n matrix B .
On exit: if $m - k - l < 0$, $B(m - k + 1 : l, n + m - k - l + 1 : n)$ contains the submatrix R_{33} of R .
- 12: LDB – INTEGER Input
On entry: the first dimension of the array B as declared in the (sub)program from which F08YSF (ZTGSJA) is called.
Constraint: $LDB \geq \max(1, P)$.
- 13: TOLA – **double precision** Input
14: TOLB – **double precision** Input
On entry: TOLA and TOLB are the convergence criteria for the Jacobi–Kogbetliantz iteration procedure. Generally, they should be the same as used in the preprocessing step performed by F08VSF (ZGGSVP), say
- $$\begin{aligned} \text{TOLA} &= \max(M, N) \|A\| \epsilon, \\ \text{TOLB} &= \max(P, N) \|B\| \epsilon, \end{aligned}$$
- where ϵ is the **machine precision**.
- 15: ALPHA(N) – **double precision** array Output
On exit: see the description of BETA.
- 16: BETA(N) – **double precision** array Output
On exit: ALPHA and BETA contain the generalized singular value pairs of A and B :
 $\text{ALPHA}(i) = 1, \text{BETA}(i) = 0$, for $i = 1, 2, \dots, k$, and
if $m - k - l \geq 0$, $\text{ALPHA}(i) = \alpha_i, \text{BETA}(i) = \beta_i$, for $i = k + 1, k + 2, \dots, k + l$, or
if $m - k - l < 0$, $\text{ALPHA}(i) = \alpha_i, \text{BETA}(i) = \beta_i$, for $i = k + 1, k + 2, \dots, m$ and
 $\text{ALPHA}(i) = 0, \text{BETA}(i) = 1$, for $i = m + 1, m + 2, \dots, k + l$.
Furthermore, if $k + l < n$, $\text{ALPHA}(i) = \text{BETA}(i) = 0$, for $i = k + l + 1, k + l + 2, \dots, n$.
- 17: U(LDU,*) – **complex*16** array Input/Output
Note: the second dimension of the array U must be at least $\max(1, M)$.
On entry: if JOBU = 'U', U must contain an m by m matrix U_1 (usually the unitary matrix returned by F08VSF (ZGGSVP)).
On exit: if JOBU = 'T', U contains the unitary matrix U .
If JOBU = 'U', U contains the product $U_1 U$.
If JOBU = 'N', U is not referenced.
- 18: LDU – INTEGER Input
On entry: the first dimension of the array U as declared in the (sub)program from which F08YSF (ZTGSJA) is called.
Constraints:
if JOBU = 'U', $LDU \geq \max(1, M)$;
otherwise $LDU \geq 1$.

- 19: $V(LDV,*)$ – **complex*16** array *Input/Output*
Note: the second dimension of the array V must be at least $\max(1, P)$.
On entry: if $JOBV = 'V'$, V must contain an p by p matrix V_1 (usually the unitary matrix returned by F08VSF (ZGGSPV)).
On exit: if $JOBV = 'I'$, V contains the unitary matrix V .
 If $JOBV = 'V'$, V contains the product $V_1 V$.
 If $JOBV = 'N'$, V is not referenced.
- 20: LDV – INTEGER *Input*
On entry: the first dimension of the array V as declared in the (sub)program from which F08YSF (ZTGSJA) is called.
Constraints:
 if $JOBV = 'V'$, $LDV \geq \max(1, P)$;
 otherwise $LDV \geq 1$.
- 21: $Q(LDQ,*)$ – **complex*16** array *Input/Output*
Note: the second dimension of the array Q must be at least $\max(1, N)$.
On entry: if $JOBQ = 'Q'$, Q must contain an n by n matrix Q_1 (usually the unitary matrix returned by F08VSF (ZGGSPV)).
On exit: if $JOBQ = 'I'$, Q contains the unitary matrix Q .
 If $JOBQ = 'Q'$, Q contains the product $Q_1 Q$.
 If $JOBQ = 'N'$, Q is not referenced.
- 22: LDQ – INTEGER *Input*
On entry: the first dimension of the array Q as declared in the (sub)program from which F08YSF (ZTGSJA) is called.
Constraints:
 if $JOBQ = 'Q'$, $LDQ \geq \max(1, N)$;
 otherwise $LDQ \geq 1$.
- 23: $WORK(2 \times N)$ – **complex*16** array *Workspace*
- 24: $NCYCLE$ – INTEGER *Output*
On exit: the number of cycles required for convergence.
- 25: $INFO$ – 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 $INFO = -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

$INFO = 1$

The procedure does not converge after 40 cycles.

7 Accuracy

The computed generalized singular value decomposition is nearly the exact generalized singular value decomposition for nearby matrices $(A + E)$ and $(B + F)$, where

$$\|E\|_2 = O\epsilon\|A\|_2 \quad \text{and} \quad \|F\|_2 = O\epsilon\|B\|_2,$$

and ϵ is the *machine precision*. See Section 4.12 of Anderson *et al.* (1999) for further details.

8 Further Comments

The real analogue of this routine is F08YEF (DTGSJA).

9 Example

This example finds the generalized singular value decomposition

$$A = U\Sigma_1(0 \ R)Q^H, \quad B = V\Sigma_2(0 \ R)Q^H,$$

of the matrix pair (A, B) , where

$$A = \begin{pmatrix} 0.96 - 0.81i & -0.03 + 0.96i & -0.91 + 2.06i & -0.05 + 0.41i \\ -0.98 + 1.98i & -1.20 + 0.19i & -0.66 + 0.42i & -0.81 + 0.56i \\ 0.62 - 0.46i & 1.01 + 0.02i & 0.63 - 0.17i & -1.11 + 0.60i \\ 0.37 + 0.38i & 0.19 - 0.54i & -0.98 - 0.36i & 0.22 - 0.20i \\ 0.83 + 0.51i & 0.20 + 0.01i & -0.17 - 0.46i & 1.47 + 1.59i \\ 1.08 - 0.28i & 0.20 - 0.12i & -0.07 + 1.23i & 0.26 + 0.26i \end{pmatrix}$$

and

$$B = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix}.$$

9.1 Program Text

```
*      F08YSF Example Program Text
*      Mark 21 Release. NAG Copyright 2004.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX, PMAX
      PARAMETER        (MMAX=10,NMAX=10,PMAX=10)
      INTEGER          LDA, LDB, LDQ, LDU, LDV
      PARAMETER        (LDA=MMAX,LDB=PMAX,LDQ=NMAX,LDU=MMAX,LDV=PMAX)
*      .. Local Scalars ..
      DOUBLE PRECISION EPS, TOLA, TOLB
      INTEGER          I, IFAIL, INFO, IRANK, J, K, L, M, N, NCYCLE, P
*      .. Local Arrays ..
      COMPLEX *16      A(LDA,NMAX), B(LDB,NMAX), Q(LDQ,NMAX), TAU(NMAX),
+      U(LDU,MMAX), V(LDV,PMAX), WORK(MMAX+3*NMAX+PMAX)
      DOUBLE PRECISION ALPHA(NMAX), BETA(NMAX), RWORK(2*NMAX)
      INTEGER          IWORK(NMAX)
      CHARACTER        CLABS(1), RLABS(1)
*      .. External Functions ..
      DOUBLE PRECISION F06UAF, X02AJF
      EXTERNAL          F06UAF, X02AJF
*      .. External Subroutines ..
      EXTERNAL          X04DBF, ZGGSVP, ZTGSJA
*      .. Intrinsic Functions ..
      INTRINSIC          MAX
*      .. Executable Statements ..
      WRITE (NOUT,*) 'F08YSF Example Program Results'
      WRITE (NOUT,*)
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) M, N, P
      IF (M.LE.MMAX .AND. N.LE.NMAX .AND. P.LE.PMAX) THEN
```

```

*
*      Read the m by n matrix A and p by n matrix B from data file
*
*      READ (NIN,*) ((A(I,J),J=1,N),I=1,M)
*      READ (NIN,*) ((B(I,J),J=1,N),I=1,P)
*
*      Compute TOLA and TOLB as
*      TOLA = max(M,N)*norm(A)*macheps
*      TOLB = max(P,N)*norm(B)*macheps
*
*      EPS = X02AJF()
*      TOLA = MAX(M,N)*F06UAF('One-norm',M,N,A,LDA,RWORK)*EPS
*      TOLB = MAX(P,N)*F06UAF('One-norm',P,N,B,LDB,RWORK)*EPS
*
*      Compute the factorization of (A, B)
*      (A = U1*S*(Q1**H), B = V1*T*(Q1**H))
*
*      CALL ZGGSVP('U','V','Q',M,P,N,A,LDA,B,LDB,TOLA,TOLB,K,L,U,LDU,
+          V,LDV,Q,LDQ,IWORK,RWORK,TAU,WORK,INFO)
*
*      Compute the generalized singular value decomposition of (A, B)
*      (A = U*D1*(O R)*(Q**H), B = V*D2*(O R)*(Q**H))
*
*      CALL ZTGSJA('U','V','Q',M,P,N,K,L,A,LDA,B,LDB,TOLA,TOLB,ALPHA,
+          BETA,U,LDU,V,LDV,Q,LDQ,WORK,NCYCLE,INFO)
*
*      IF (INFO.EQ.0) THEN
*
*          Print solution
*
*          IRANK = K + L
*          WRITE (NOUT,*)
+          'Number of infinite generalized singular values (K)'
*          WRITE (NOUT,99999) K
*          WRITE (NOUT,*)
+          'Number of finite generalized singular values (L)'
*          WRITE (NOUT,99999) L
*          WRITE (NOUT,*)
+          'Effective Numerical rank of (A**T B**T)**T (K+L)'
*          WRITE (NOUT,99999) IRANK
*          WRITE (NOUT,*)
*          WRITE (NOUT,*) 'Finite generalized singular values'
*          WRITE (NOUT,99998) (ALPHA(J)/BETA(J),J=K+1,IRANK)
*
*          IFAIL = 0
*          WRITE (NOUT,*)
+          CALL X04DBF('General',' ',M,M,U,LDU,'Bracketed','1P,E12.4',
+              'Orthogonal matrix U','Integer',RLABS,'Integer',
+              CLABS,80,0,IFAIL)
*          WRITE (NOUT,*)
+          CALL X04DBF('General',' ',P,P,V,LDV,'Bracketed','1P,E12.4',
+              'Orthogonal matrix V','Integer',RLABS,'Integer',
+              CLABS,80,0,IFAIL)
*          WRITE (NOUT,*)
+          CALL X04DBF('General',' ',N,N,Q,LDQ,'Bracketed','1P,E12.4',
+              'Orthogonal matrix Q','Integer',RLABS,'Integer',
+              CLABS,80,0,IFAIL)
*          WRITE (NOUT,*)
+          CALL X04DBF('Upper triangular','Non-unit',IRANK,IRANK,
+              A(1,N-IRANK+1),LDA,'Bracketed','1P,E12.4',
+              'Non singular upper triangular matrix R',
+              'Integer',RLABS,'Integer',CLABS,80,0,IFAIL)
*          WRITE (NOUT,*)
*          WRITE (NOUT,*) 'Number of cycles of the Kogbetliantz method'
*          WRITE (NOUT,99999) NCYCLE
*      ELSE
*          WRITE (NOUT,99997) 'Failure in ZTGSJA. INFO =', INFO
*      END IF
*  ELSE
*      WRITE (NOUT,*) 'MMAX and/or NMAX too small'
*  END IF

```

```

*
99999 FORMAT (1X,I5)
99998 FORMAT (3X,8(1P,E12.4))
99997 FORMAT (1X,A,I4)
      END

```

9.2 Program Data

F08YSF Example Program Data

```

      6              4              2              :Values of M, N and P

( 0.96,-0.81) (-0.03, 0.96) (-0.91, 2.06) (-0.05, 0.41)
(-0.98, 1.98) (-1.20, 0.19) (-0.66, 0.42) (-0.81, 0.56)
( 0.62,-0.46) ( 1.01, 0.02) ( 0.63,-0.17) (-1.11, 0.60)
( 0.37, 0.38) ( 0.19,-0.54) (-0.98,-0.36) ( 0.22,-0.20)
( 0.83, 0.51) ( 0.20, 0.01) (-0.17,-0.46) ( 1.47, 1.59)
( 1.08,-0.28) ( 0.20,-0.12) (-0.07, 1.23) ( 0.26, 0.26) :End of matrix A

( 1.00, 0.00) ( 0.00, 0.00) (-1.00, 0.00) ( 0.00, 0.00)
( 0.00, 0.00) ( 1.00, 0.00) ( 0.00, 0.00) (-1.00, 0.00) :End of matrix B

```

9.3 Program Results

F08YSF Example Program Results

Number of infinite generalized singular values (K)

2

Number of finite generalized singular values (L)

2

Effective Numerical rank of (A**T B**T)**T (K+L)

4

Finite generalized singular values

2.0720E+00 1.1058E+00

Orthogonal matrix U

```

      1              2
1 ( -1.3038E-02, -3.2595E-01) ( -1.4039E-01, -2.6167E-01)
2 (  4.2764E-01, -6.2582E-01) (  8.6298E-02, -3.8174E-02)
3 ( -3.2595E-01,  1.6428E-01) (  3.8163E-01, -1.8219E-01)
4 (  1.5906E-01, -5.2151E-03) ( -2.8207E-01,  1.9732E-01)
5 ( -1.7210E-01, -1.3038E-02) ( -5.0942E-01, -5.0319E-01)
6 ( -2.6336E-01, -2.4772E-01) ( -1.0861E-01,  2.8474E-01)

      3              4
1 (  2.5177E-01, -7.9789E-01) ( -5.0956E-02, -2.1750E-01)
2 ( -3.2188E-01,  1.6112E-01) (  1.1979E-01,  1.6319E-01)
3 (  1.3231E-01, -1.4565E-02) ( -5.0671E-01,  1.8615E-01)
4 (  2.1598E-01,  1.8813E-01) ( -4.0163E-01,  2.6787E-01)
5 (  3.6488E-02,  2.0316E-01) (  1.9271E-01,  1.5574E-01)
6 (  1.0906E-01, -1.2712E-01) ( -8.8159E-02,  5.6169E-01)

      5              6
1 ( -4.5947E-02,  1.4052E-04) ( -5.2773E-02, -2.2492E-01)
2 ( -8.0311E-02, -4.3605E-01) ( -3.8117E-02, -2.1907E-01)
3 (  5.9714E-02, -5.8974E-01) ( -1.3850E-01, -9.0941E-02)
4 ( -4.6443E-02,  3.0864E-01) ( -3.7354E-01, -5.5148E-01)
5 (  5.7843E-01, -1.2439E-01) ( -1.8815E-02, -5.5686E-02)
6 (  1.5763E-02,  4.7130E-02) (  6.5007E-01,  4.9173E-03)

```

Orthogonal matrix V

```

      1              2
1 (  9.8930E-01,  6.5610E-19) ( -1.1461E-01,  9.0250E-02)
2 ( -1.1461E-01, -9.0250E-02) ( -9.8930E-01,  6.5610E-19)

```

Orthogonal matrix Q

```

      1              2
1 (  7.0711E-01,  0.0000E+00) (  0.0000E+00,  0.0000E+00)
2 (  0.0000E+00,  0.0000E+00) (  7.0711E-01,  0.0000E+00)

```

```

3  (  7.0711E-01,  0.0000E+00) (  0.0000E+00,  0.0000E+00)
4  (  0.0000E+00,  0.0000E+00) (  7.0711E-01,  0.0000E+00)

```

```

3 4
1  (  6.9954E-01,  8.0203E-19) (  8.1044E-02, -6.3817E-02)
2  ( -8.1044E-02, -6.3817E-02) (  6.9954E-01, -8.0203E-19)
3  ( -6.9954E-01, -8.0203E-19) ( -8.1044E-02,  6.3817E-02)
4  (  8.1044E-02,  6.3817E-02) ( -6.9954E-01,  8.0203E-19)

```

Non singular upper triangular matrix R

```

1 2
1  ( -2.7118E+00,  0.0000E+00) ( -1.4390E+00, -1.0315E+00)
2  ( -1.8583E+00,  0.0000E+00)
3
4

```

```

3 4
1  ( -7.6930E-02,  1.3613E+00) ( -2.8137E-01, -3.2425E-02)
2  ( -1.0760E+00,  3.1016E-02) (  1.3292E+00,  3.6772E-01)
3  (  3.2537E+00,  0.0000E+00) ( -6.3858E-17,  6.3858E-17)
4  ( -2.1084E+00,  0.0000E+00)

```

Number of cycles of the Kogbetliantz method

2
