```
!.....................................................................................................................!:.:.:
!.......................... File: matrix_mod_common.f90 ...............................................................!:.:.:
module              matrix_mod_common                                                                               !:.:.:
 interface          NaN                                                                                             !:.:.:
 interface          isNaN                                                                                           !:.:.:
 subroutine         matrix_random_init                                                                              !:.:.:
 subroutine         matrix_error                                  (message)                                         !:.:.:
 pure function      matrix_return_nan_complex8                    ()                      result(r)                 !:.:.:
 pure function      matrix_return_nan_complex8_z                  (z)                     result(r)                 !:.:.:
 pure function      matrix_return_nan_real8_d                     (z)                     result(r)                 !:.:.:
 pure function      matrix_return_is_nan_complex8_z               (z)                     result(itis)              !:.:.:
 pure function      matrix_return_is_nan_real8_d                  (z)                     result(itis)              !:.:.:
!.....................................................................................................................!:.:.:
!.......................... File: matrix_mod_array.f90 ................................................................!:.:.:
module              matrix_mod_array                                                                                !:.:.:
 interface          operator(.mm.)                                                                                  !:.:.:
 interface          random_number                                                                                   !:.:.:
 interface          mmmult                                                                                          !:.:.:
 interface          mvmult                                                                                          !:.:.:
 interface          vmmult                                                                                          !:.:.:
 interface          lmatmul                                                                                         !:.:.:
 interface          eigenvalues                                                                                     !:.:.:
 interface          eigenvectors                                                                                    !:.:.:
 interface          determinant                                                                                     !:.:.:
 interface          lndet                                                                                           !:.:.:
 interface          pfaffian                                                                                        !:.:.:
 interface          lnPfaffian                                                                                      !:.:.:
 interface          inverse                                                                                         !:.:.:
 interface          norm                                                                                            !:.:.:
 interface          hermitian                                                                                       !:.:.:
 interface          hermitian_set                                                                                   !:.:.:
 interface          symmetric                                                                                       !:.:.:
 interface          symmetric_set                                                                                   !:.:.:
 interface          antisymmetric                                                                                   !:.:.:
 interface          antisymmetric_set                                                                               !:.:.:
 interface          diagonal                                                                                        !:.:.:
 interface          diagonalMatrix                                                                                  !:.:.:
 interface          identitymatrix                                                                                  !:.:.:
 interface          cidentitymatrix                                                                                  !:.:.
 interface          didentitymatrix                                                                                  !:.:.
 interface          paulimatrix                                                                                     !:.:.:
 interface          trace                                                                                           !:.:.:
 interface          trace2                                                                                          !:.:.:
 interface          trace2c                                                                                         !:.:.:
 interface          traceless                                                                                       !:.:.:
 interface          traceless_set                                                                                   !:.:.:
 interface          isHermitian                                                                                     !:.:.:
 interface          isSymmetric                                                                                     !:.:.:
 interface          isAntiSymmetric                                                                                 !:.:.:
 interface          sort                                                                                            !:.:.:
 interface          print                                                                                           !:.:.:
 interface          printna                                                                                         !:.:.:
```

```
 interface          save                                                                                    !:.:.:
 interface          read                                                                                    !:.:.:
 interface          isNaN                                                                                   !:.:.:
!......................... array2 matrix functions ..................................................................!:.:.:
 pure function      array2_trace2_connected                    (C,mtype)              result(t)             !:.:.:
 pure function      array2_trace2_connected_d                  (C,mtype)              result(t)             !:.:.:
 pure function      array2_trace2                              (C,mtype)              result(t)             !:.:.:
 pure function      array2_trace2_d                            (C,mtype)              result(t)             !:.:.:
 pure function      array2_traceless_get                       (C)                    result(B)             !:.:.:
 pure function      array2_traceless_get_d                     (C)                    result(B)             !:.:.:
 pure subroutine    array2_traceless_set                       (C)                                          !:.:.:
 pure subroutine    array2_traceless_set_d                     (C)                                          !:.:.:
 pure function      array2_trace                               (C)                    result(t)             !:.:.:
 pure function      array2_trace_d                             (C)                    result(t)             !:.:.:
 pure function      array2_diagonal_get                        (C)                    result(d)             !:.:.:
 pure function      array2_diagonal_get_d                      (C)                    result(d)             !:.:.:
 pure function      array2_diagonal_set                        (d)                    result(C)             !:.:.:
 pure function      array2_diagonal_set_d                      (d)                    result(C)             !:.:.:
 pure function      array2_diagonal_set_from_real              (r,n)                  result(C)             !:.:.:
 pure function      array2_diagonal_set_from_real_d            (r,n)                  result(C)             !:.:.:
 pure function      array2_diagonal_set_from_complex           (z,n)                  result(C)             !:.:.:
 pure function      array2_diagonal_set_from_complex_matrix    (C)                    result(D)             !:.:.:
 pure function      array2_diagonal_set_from_real_matrix_d     (C)                    result(D)             !:.:.:
 pure function      array2_diagonal_set_identity_complex_matrix (n)                   result(C)             !:.:.:
 pure function      array2_diagonal_set_identity_real_matrix   (n)                    result(C)             !:.:.:
 pure function      array2_is_Hermitian                        (C)                    result(r)             !:.:.:
 pure function      array2_is_Symmetric                        (C)                    result(r)             !:.:.:
 pure function      array2_is_Symmetric_d                      (C)                    result(r)             !:.:.:
 pure function      array2_is_AntiSymmetric                    (C)                    result(r)             !:.:.:
 pure function      array2_is_AntiSymmetric_d                  (C)                    result(r)             !:.:.:
 pure function      array2_PauliMatrix                         (n)                    result(C)             !:.:.:
!......................... array2 procedures.........................................................................!:.:.:
 pure function      array3_norm                                (C)                    result(r)             !:.:.:
 pure function      array3_norm_d                              (C)                    result(r)             !:.:.:
 pure function      array2_norm                                (C)                    result(r)             !:.:.:
 pure function      array2_norm_d                              (C)                    result(r)             !:.:.:
 pure function      array1_norm                                (C)                    result(r)             !:.:.:
 pure function      array1_norm_d                              (C)                    result(r)             !:.:.:
 pure subroutine    array2_hermitian_set                       (C,uplo)                                     !:.:.:
 pure function      array2_hermitian_get                       (C)                    result(CH)            !:.:.:
 pure subroutine    array2_symmetric_set                       (C,uplo)                                     !:.:.:
 pure function      array2_symmetric_get                       (C)                    result(CS)            !:.:.:
 pure subroutine    array2_symmetric_set_d                     (C,uplo)                                     !:.:.:
 pure function      array2_symmetric_get_d                     (C)                    result(CS)            !:.:.:
 pure subroutine    array2_antisymmetric_set                   (C,uplo)                                     !:.:.:
 pure function      array2_antisymmetric_get                   (C)                    result(CS)            !:.:.:
 pure subroutine    array2_antisymmetric_set_d                 (C,uplo)                                     !:.:.:
 pure function      array2_antisymmetric_get_d                 (C)                    result(CS)            !:.:.:
 subroutine         array2_gauss_set                           (C,sigma)                                    !:.:.:
 subroutine         array2_gauss_set_d                         (C,sigma)                                    !:.:.:
 subroutine         array2_random_set                          (C)                                          !:.:.:
 subroutine         array2_random_set_d                        (C)                                          !:.:.:
```

```
 function           array1_sort                                  (C,by)                    result(D)                 !:.:.:
 function           array1_sort_d                                (C,by)                    result(D)                 !:.:.:
 recursive subroutine array1_quicksortZbyModulus                 (A,first,last)                                      !:.:.:
 recursive subroutine array1_reversequicksortZbyModulus          (A,first,last)                                      !:.:.:
 subroutine         array1_sortZbyModulus                        (C)                                                 !:.:.:
 recursive subroutine array1_quicksortZbyRealPart                (A,first,last)                                      !:.:.:
 recursive subroutine array1_reversequicksortZbyRealPart         (A,first,last)                                      !:.:.:
 recursive subroutine array1_quicksortZbyImagPart                (A,first,last)                                      !:.:.:
 recursive subroutine array1_reversequicksortZbyImagPart         (A,first,last)                                      !:.:.:
 recursive subroutine array1_quicksortDbyModulus_d               (A,first,last)                                      !:.:.:
 recursive subroutine array1_reversequicksortDbyModulus_d        (A,first,last)                                      !:.:.:
 recursive subroutine array1_quicksortDbyValue_d                 (A,first,last)                                      !:.:.:
 recursive subroutine array1_reversequicksortDbyValue_d          (A,first,last)                                      !:.:.:
!......................... random_number interface: ..........................................................!:.:.:
 subroutine         random_number_complex_scalar                 (z)                                                 !:.:.:
 subroutine         random_number_array3                         (C)                                                 !:.:.:
 subroutine         random_number_array2                         (C)                                                 !:.:.:
 subroutine         random_number_array1                         (C)                                                 !:.:.:
 subroutine         random_number_complex_scalar_gaussian        (z,sigma)                                           !:.:.:
 subroutine         random_number_real_scalar_gaussian           (r,sigma)                                           !:.:.:
 subroutine         random_number_array3_gaussian                (C,sigma)                                           !:.:.:
 subroutine         random_number_array3_gaussian_d              (C,sigma)                                           !:.:.:
 subroutine         random_number_array2_gaussian                (C,sigma)                                           !:.:.:
 subroutine         random_number_array2_gaussian_d              (C,sigma)                                           !:.:.:
 subroutine         random_number_array1_gaussian                (C,sigma)                                           !:.:.:
 subroutine         random_number_array1_gaussian_d              (C,sigma)                                           !:.:.:
!......................... array2 matmul operator overload ...................................................!:.:.:
 pure function      array2_matmul_array2                         (C1,C2)                   result(C3)                !:.:.:
 pure function      array2_matmul_array2_d                       (C1,C2)                   result(C3)                !:.:.:
 pure function      array2_d_matmul_array2                       (C1,C2)                   result(C3)                !:.:.:
 pure function      array2_d_matmul_array2_d                     (C1,C2)                   result(C3)                !:.:.:
 pure function      array2_matmul_array1                         (C1,v2)                   result(v3)                !:.:.:
 pure function      array1_matmul_array2                         (v1,C2)                   result(v3)                !:.:.:
 pure function      array2_d_matmul_array1_d                     (C1,v2)                   result(v3)                !:.:.:
 pure function      array1_d_matmul_array2_d                     (v1,C2)                   result(v3)                !:.:.:
 pure function      array2_matmul_array1_d                       (C1,v2)                   result(v3)                !:.:.:
 pure function      array1_d_matmul_array2                       (v1,C2)                   result(v3)                !:.:.:
 pure function      array2_d_matmul_array1                       (C1,v2)                   result(v3)                !:.:.:
 pure function      array1_matmul_array2_d                       (v1,C2)                   result(v3)                !:.:.:
!......................... utilities: ........................................................................!:.:.:
 subroutine         array2_print                                 (C,unit,fmt,form,name,ips,ipe,jps,jpe)              !:.:.:
 subroutine         array1_print                                 (C,unit,fmt,form,name,ips,ipe)                      !:.:.:
 subroutine         array2_print_d                               (C,unit,fmt,form,name,ips,ipe,jps,jpe)              !:.:.:
 subroutine         array1_print_d                               (C,unit,fmt,form,name,ips,ipe)                      !:.:.:
 subroutine         array2_print_nonallocatable                  (C,unit,fmt,form,name,ips,ipe,jps,jpe)              !:.:.:
 subroutine         array2_print_nonallocatable_d                (C,unit,fmt,form,name,ips,ipe,jps,jpe)              !:.:.:
 subroutine         array3_save_matrix                           (C,unit,fmt)                                        !:.:.:
 subroutine         array2_save_matrix                           (C,unit,fmt)                                        !:.:.:
 subroutine         array1_save_matrix                           (C,unit,fmt)                                        !:.:.:
 subroutine         array3_save_matrix_d                         (C,unit,fmt)                                        !:.:.:
 subroutine         array2_save_matrix_d                         (C,unit,fmt)                                        !:.:.:
 subroutine         array1_save_matrix_d                         (C,unit,fmt)                                        !:.:.:
```

```
 subroutine         array3_read_matrix                                (C,unit)                                                !:.:.:
 subroutine         array2_read_matrix                                (C,unit)                                                !:.:.:
 subroutine         array1_read_matrix                                (C,unit)                                                !:.:.:
 subroutine         array3_read_matrix_d                              (C,unit)                                                !:.:.:
 subroutine         array2_read_matrix_d                              (C,unit)                                                !:.:.:
 subroutine         array1_read_matrix_d                              (C,unit)                                                !:.:.:
 pure function      array3_is_nan                                     (C)                    result(itis)                     !:.:.:
 pure function      array2_is_nan                                     (C)                    result(itis)                     !:.:.:
 pure function       array1_is_nan                                    (C)                    result(itis)                     !:.:.:
 pure function      array3_is_nan_d                                   (C)                    result(itis)                     !:.:.:
 pure function      array2_is_nan_d                                   (C)                    result(itis)                     !:.:.:
 pure function      array1_is_nan_d                                   (C)                    result(itis)                     !:.:.:
!.....................................................................................................................!:.:.:
!.......................... File: matrix_mod_matrix.f90 ................................................................!:.:.:
!.....................................................................................................................!:.:.:
module              matrix_mod_matrix                                                                                    !:.:.:
 use                matrix_mod_common                                                                                    !:.:.:
 use                matrix_mod_array                                                                                     !:.:.:
!---------------------------------------------------------------------------------------------------------------------!:.:.:
 public                                  :: Matrix, DMatrix, Vector, DVector                                          !:.:.:
 public                                  :: mmmult, mvmult , vmmult                                                   !:.:.:
 public                                  :: random_number, mcmplx, real, aimag, conjg, transpose, hermitian, norm, symmetric  !:.:.:
 public                                  :: isNaN, sort, dot_product, maxval, minval, trace, trace2, trace2c, traceless  !:.:.:
 public                                  :: diagonal, diagonalMatrix, metadata_copy, inverse, determinant, eigenvalues, eigenvectors !:.:.:
 public                                  :: lndet, Pfaffian, lnPfaffian, isHermitian, isSymmetric, isAntiSymmetric    !:.:.:
 public                                  :: traceless_set, hermitian_set, symmetric_set, antisymmetric_set            !:.:.:
 public                                  :: abs, sin, cos, exp, log, sqrt                                             !:.:.:
 public                                  :: matrix_random_init, NaN, f_mout                                           !:.:.:
 public                                  :: assignment(=), operator(+), operator(-), operator(*), operator(/) , operator(**)  !:.:.:
!---------------------------------------------------------------------------------------------------------------------!:.:.:
 type               MatrixClass                                                                                       !:.:.:
  integer                                :: m=0, n =0                                                                 !:.:.:
  integer                                :: is=0,ie=0,js=0,je=0                                                       !:.:.:
  character(mtype_len)                   :: mtype='GG'                                                                !:.:.:
  character( name_len)                   :: name=''                                                                   !:.:.:
  generic                                :: save           => matrix_save                                             !:.:.:
  generic                                :: read           => matrix_read                                             !:.:.:
  generic                                :: print          => matrix_print                                            !:.:.:
  generic                                :: random         => matrix_random_set                                       !:.:.:
  generic                                :: gaussian       => matrix_gaussian_set                                     !:.:.:
 type,      extends(MatrixClass)        :: Matrix                                                                    !:.:.:
  complex(dp), allocatable               :: v(:,:)                                                                    !:.:.:
  generic                                :: hermitian_set  => matrix_hermitian_set                                    !:.:.:
  generic                                :: conjg          => matrix_return_conjg                                     !:.:.:
  generic                                :: transpose      => matrix_return_transpose                                 !:.:.:
  generic                                :: hermitian      => matrix_return_hermitian                                 !:.:.:
  generic                                :: dagger         => matrix_return_hermitian                                 !:.:.:
  generic                                :: re             => matrix_return_real_dmatrix                              !:.:.:
  generic                                :: im             => matrix_return_imag_dmatrix                              !:.:.:
 type,      extends(MatrixClass)        :: DMatrix                                                                   !:.:.:
  real   (dp), allocatable               :: v(:,:)                                                                    !:.:.:
  generic                                :: symmetric_set  => matrix_symmetric_set_d                                  !:.:.:
```

```
 generic                             :: transpose      => matrix_return_transpose_d                !:.:.:
 generic                             :: symmetric      => matrix_return_transpose_d                !:.:.:
type               Vector                                                                           !:.:.:
 integer                             :: n =0                                                        !:.:.:
 integer                             :: is=0,ie=0                                                   !:.:.:
 character( name_len)                :: name=''                                                     !:.:.:
type               DVector                                                                          !:.:.:
 integer                             :: n =0                                                        !:.:.:
 integer                             :: is=0,ie=0                                                   !:.:.:
 character( name_len)                :: name=''                                                     !:.:.:
interface          assignment(=)                                                                    !:.:.:
interface          operator(+)                                                                      !:.:.:
interface          operator(-)                                                                      !:.:.:
interface          operator(*)                                                                      !:.:.:
interface          operator(/)                                                                      !:.:.:
interface          operator(**)                                                                     !:.:.:
interface          random_number                                                                    !:.:.:
interface          mmmult                                                                           !:.:.:
interface          mvmult                                                                           !:.:.:
interface          vmmult                                                                           !:.:.:
interface          conjg                                                                            !:.:.:
interface          transpose                                                                        !:.:.:
interface          hermitian                                                                        !:.:.:
interface          symmetric                                                                        !:.:.:
interface          trace                                                                            !:.:.:
interface          trace2                                                                           !:.:.:
interface          trace2c                                                                          !:.:.:
interface          inverse                                                                          !:.:.:
interface          eigenvalues                                                                      !:.:.:
interface          eigenvectors                                                                     !:.:.:
interface          determinant                                                                      !:.:.:
interface          lndet                                                                            !:.:.:
interface          Pfaffian                                                                         !:.:.:
interface          lnPfaffian                                                                       !:.:.:
interface          diagonal                                                                         !:.:.:
interface          diagonalMatrix                                                                   !:.:.:
interface          sort                                                                             !:.:.:
interface          norm                                                                             !:.:.:
interface          real                                                                             !:.:.:
interface          aimag                                                                            !:.:.:
interface          mcmplx                                                                           !:.:.:
interface          dot_product                                                                      !:.:.:
interface          maxval                                                                           !:.:.:
interface          minval                                                                           !:.:.:
interface          traceless                                                                        !:.:.:
interface          traceless_set                                                                    !:.:.:
interface          hermitian_set                                                                    !:.:.:
interface          symmetric_set                                                                    !:.:.:
interface          antisymmetric_set                                                                !:.:.:
interface          isHermitian                                                                      !:.:.:
interface          isSymmetric                                                                      !:.:.:
interface          isAntiSymmetric                                                                  !:.:.:
```

```
 interface          isNaN                                                                                                                !:.:.:
 interface          metadata_copy                                                                                                        !:.:.:
 interface          abs                                                                                                                  !:.:.:
 interface          sin                                                                                                                  !:.:.:
 interface          cos                                                                                                                  !:.:.:
 interface          exp                                                                                                                  !:.:.:
 interface          log                                                                                                                  !:.:.:
 interface          sqrt                                                                                                                 !:.:.:
!.....................................................................................................................................!:.:.:
!.......................... File: tensorprod_mod.f90   ..............................................................................!:.:.:
module              tensorprod_mod                                                                                                        !:
!-------------------------------------------------------------------------------------------------------------------------------------!:.:.:
 public                                 :: tensorprod, vec2row, vec2col, vectorize                                                    !:.:.:
!-------------------------------------------------------------------------------------------------------------------------------------!:.:.:
 interface          tensorprod                                                                                                           !:.:.:
 pure function      tensorprod_complex_complex                              (X,Y)                   result(Z)                            !:.:.:
 pure function      tensorprod_complex_3                                    (X1,X2,X3)              result(Z)                            !:.:.:
 pure function      tensorprod_complex_4                                    (X1,X2,X3,X4)           result(Z)                            !:.:.:
 pure function      tensorprod_complex_5                                    (X1,X2,X3,X4,X5)        result(Z)                            !:.:.:
 pure function      tensorprod_complex_6                                    (X1,X2,X3,X4,X5,X6)     result(Z)                            !:.:.:
 pure function      tensorprod_real_real                                    (X,Y)                   result(Z)                            !:.:.:
 pure function      tensorprod_real_3                                       (X1,X2,X3)              result(Z)                            !:.:.:
 pure function      tensorprod_real_4                                       (X1,X2,X3,X4)           result(Z)                            !:.:.:
 pure function      tensorprod_real_5                                       (X1,X2,X3,X4,X5)        result(Z)                            !:.:.:
 pure function      tensorprod_real_6                                       (X1,X2,X3,X4,X5,X6)     result(Z)                            !:.:.:
 pure function      tensorprod_complex_vec_vec                              (u,v)                   result(Z)                            !:.:.:
 pure function      tensorprod_real_vec_vec                                 (u,v)                   result(Z)                            !:.:.:
 pure function      vec2col_complex                                         (u)                     result(Z)                            !:.:.:
 pure function      vec2row_complex                                         (u)                     result(Z)                            !:.:.:
 pure function      vec2col_real                                            (u)                     result(Z)                            !:.:.:
 pure function      vec2row_real                                            (u)                     result(Z)                            !:.:.:
 pure function      vectorize_complex                                       (Z)                     result(u)                            !:.:.:
 pure function      vectorize_real                                          (Z)                     result(u)                            !:.:.:
!.....................................................................................................................................!:.:.:
!.......................... File: array_mod.f90        ..............................................................................!:.:.:
module              array_mod                                                                                                            !:.:.:
!-------------------------------------------------------------------------------------------------------------------------------------!:.:.:
 public                                 :: mmmult , mvmult, vmmult                                                                    !:.:.:
 public                                 :: lmatmul, eigenvalues, eigenvectors, determinant, lndet, pfaffian, lnPfaffian, inverse      !:.:.:
 public                                 :: hermitian, hermitian_set, symmetric, symmetric_set, antisymmetric, antisymmetric_set       !:.:.:
 public                                 :: diagonal, diagonalMatrix, trace, trace2, trace2c, traceless, traceless_set                 !:.:.:
 public                                 :: identitymatrix, cidentitymatrix, didentitymatrix, paulimatrix                              !:.:.:
 public                                 :: norm, isHermitian, isSymmetric, isAntisymmetric, sort                                      !:.:.:
 public                                 :: random_number, matrix_random_init                                                          !:.:.:
 public                                 :: print, printna, save, read, isNaN, NaN                                                     !:.:.:
 public                                 :: operator(.mm.)                                                                             !:.:.:
 public                                 :: tensorprod, vec2row, vec2col, vectorize                                                    !:.:.:
!-------------------------------------------------------------------------------------------------------------------------------------!:.:.:
 public                                 :: f_mout, f_minput                                                                           !:.:.:
!.....................................................................................................................................!:.:.:
!.......................... File: matrix_mod_matrix_matrix.f90 ..........................................................................!:.:.:
!.......................... type/class Matrix procedures ................................................................................!:.:.:
```

```
!.......................... Constructors ..........................................................................!:.:.:
 function           matrix_construct_zero                    (m,n,is,js,mtype,name)   result(MAT)                !:.:.:
 function           matrix_construct_array2                  (C,is,js,mtype,name)     result(MAT)                !:.:.:
 function           matrix_construct_complex                 (c,m,n,is,js,mtype,name) result(MAT)                !:.:.:
 function           matrix_construct_real                    (r,m,n,is,js,mtype,name) result(MAT)                !:.:.:
 function           matrix_construct_random                (rtype,m,n,is,js,mtype,name, sigma) result(MAT)       !:.:.:
!.......................... Components   ..........................................................................!:.:.:
 subroutine         matrix_hermitian_set                     (MAT, uplo)                                           !:.:.:
!.......................... Operators    ..........................................................................!:.:.:
 pure subroutine    matrix_assignFrom_matrix                 (MATB,MATA)                                           !:.:.:
 subroutine         matrix_assignFrom_dmatrix                (MATB,MATA)                                           !:.:.:
 subroutine         matrix_assignFrom_real                   (MATB,r)                                              !:.:.:
 subroutine         matrix_assignFrom_complex                (MATB,r)                                              !:.:.:
 subroutine         matrix_assignFrom_array2                 (MATB,C)                                              !:.:.:
 pure function      real_plus_matrix                         (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_plus_real                         (MATA,r)                 result(MATB)                 !:.:.:
 pure function      complex_plus_matrix                      (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_plus_complex                      (MATA,r)                 result(MATB)                 !:.:.:
 function           matrix_plus_array2                       (MATA,C)                 result(MATB)                 !:.:.:
 function           array2_plus_matrix                       (C,MATA)                 result(MATB)                 !:.:.:
 function           matrix_plus_matrix                       (MATA,MATB)              result(MATC)                 !:.:.:
 function           matrix_plus_dmatrix                      (MATA,MATB)              result(MATC)                 !:.:.:
 function           dmatrix_plus_matrix                      (MATA,MATB)              result(MATC)                 !:.:.:
 pure function      real_subtract_matrix                     (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_subtract_real                     (MATA,r)                 result(MATB)                 !:.:.:
 pure function      complex_subtract_matrix                  (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_subtract_complex                  (MATA,r)                 result(MATB)                 !:.:.:
 function           matrix_subtract_array2                   (MATA,C)                 result(MATB)                 !:.:.:
 function           array2_subtract_matrix                   (C,MATA)                 result(MATB)                 !:.:.:
 function           matrix_subtract_matrix                   (MATA,MATB)              result(MATC)                 !:.:.:
 function           matrix_subtract_dmatrix                  (MATA,MATB)              result(MATC)                 !:.:.:
 function           dmatrix_subtract_matrix                  (MATA,MATB)              result(MATC)                 !:.:.:
 function           matrix_return_minus_matrix               (MATA)                   result(MATB)                 !:.:.:
 pure function      real_mult_matrix                         (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_mult_real                         (MATA,r)                 result(MATB)                 !:.:.:
 pure function      complex_mult_matrix                      (r,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_mult_complex                      (MATA,r)                 result(MATB)                 !:.:.:
 pure function      matrix_mult_array2                       (MATA,C)                 result(MATB)                 !:.:.:
 pure function      array2_mult_matrix                       (C,MATA)                 result(MATB)                 !:.:.:
 pure function      matrix_mult_matrix                       (MATA,MATB)              result(MATC)                 !:.:.:
 pure function      matrix_mult_dmatrix                      (MATA,MATB)              result(MATC)                 !:.:.:
 pure function      dmatrix_mult_matrix                      (MATA,MATB)              result(MATC)                 !:.:.:
 pure function      matrix_divide_real                       (MATA,r)                 result(MATB)                 !:.:.:
 pure function      matrix_divide_complex                    (MATA,r)                 result(MATB)                 !:.:.:
 pure subroutine    matrix_mult_matrix_sub                   (MATA,MATB,MATC)                                      !:.:.:
!.......................... Linear Algebra with LAPACK ............................................................!:.:.:
 function           matrix_inverse                           (MATA)                   result(MATB)                 !:.:.:
 function           matrix_determinant                       (MAT)                    result(z)                    !:.:.:
 function           matrix_lndet                             (MAT)                    result(z)                    !:.:.:
 function           matrix_Pfaffian                          (MAT)                    result(z)                    !:.:.:
 function           matrix_lnPfaffian                        (MAT)                    result(z)                    !:.:.:
 function           matrix_eigenvalues                       (MAT)                    result(vec)                  !:.:.:
```

```
 function           matrix_eigenvectors                          (MATA,vec)              result(MATB)                    !:.:.:
!......................... Misc Procedures.....................................................................................!:.:.:
 subroutine         matrix_symmetric_set                         (MAT, uplo)                                             !:.:.:
 subroutine         matrix_antisymmetric_set                     (MAT, uplo)                                             !:.:.:
 pure function      matrix_diagonal_get                          (MAT)                   result(vec)                     !:.:.:
 function           matrix_diagonal_set_from_matrix              (MATA)                  result(MATB)                    !:.:.:
 function           matrix_diagonal_set_from_vector              (vec)                   result(MAT)                     !:.:.:
 pure function      matrix_traceless_get                         (MAT)                   result(MATB)                    !:.:.:
 pure subroutine    matrix_traceless_set                         (MAT)                                                   !:.:.:
 pure function      matrix_trace2c                               (MAT)                   result(r)                       !:.:.:
 pure function      matrix_trace2                                (MAT)                   result(r)                       !:.:.:
 pure function      matrix_trace                                 (MAT)                   result(r)                       !:.:.:
 function           matrix_norm                                  (MAT)                   result(r)                       !:.:.:
 subroutine         random_number_matrix                         (MAT)                                                   !:.:.:
 subroutine         random_number_matrix_gaussian                (MAT,sigma)                                             !:.:.:
 function           matrix_return_real_dmatrix                   (MATA)                 result(MATB)                     !:.:.:
 function           matrix_return_imag_dmatrix                   (MATA)                  result(MATB)                    !:.:.:
 function           matrix_return_conjg                          (MATA)                  result(MATB)                    !:.:.:
 function           matrix_return_transpose                      (MATA)                  result(MATB)                    !:.:.:
 function           matrix_return_hermitian                      (MATA)                  result(MATB)                    !:.:.:
 function           matrix_is_hermitian                          (MATA)                  result(r)                       !:.:.:
 function           matrix_is_symmetric                          (MATA)                  result(r)                       !:.:.:
 function           matrix_is_antisymmetric                      (MATA)                  result(r)                       !:.:.:
 function           matrix_is_nan                                (MAT)                   result(itis)                    !:.:.:
!......................... Math and Array Procedures ..........................................................................!:.:.
 pure function      matrix_abs                                   (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_sin                                   (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_cos                                   (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_exp                                   (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_log                                   (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_sqrt                                  (MAT)                   result(MATB)                    !:.:.:
 pure function      matrix_power_integer                         (MAT,n)                 result(MATB)                    !:.:.:
 pure function      matrix_power_real                            (MAT,n)                 result(MATB)                    !:.:.:
 pure function      matrix_power_complex                         (MAT,n)                 result(MATB)                    !:.:.:
!.....................................................................................................................!:.:.:
!......................... File: matrix_mod_matrix_matrixClass.f90 ....................................................!:.:.:
!......................... type/class MatrixClass procedures ..........................................................!:.:.:
!......................... Used by Constructors: ......................................................................!:.:.:
 subroutine         matrix_gaussian_set                          (MAT,sigma)                                             !:.:.:
 subroutine         matrix_random_set                            (MAT)                                                   !:.:.:
 subroutine         matrix_metadata_put                          (MAT,m,n,is,js,mtype,name)                              !:.:.:
 pure subroutine    matrix_metadata_copy(MATA,MATB)                                                                      !:.:.:
!......................... Components   ..............................................................................!:.:.:
 subroutine         matrix_read                                  (MAT,unit)                                              !:.:.:
 subroutine         matrix_save                                  (MAT,unit,fmt)                                          !:.:.:
 subroutine         matrix_print                                 (MAT,unit,fmt,form,ips,ipe,jps,jpe)                     !:.:.:
!.....................................................................................................................!:.:.:
!......................... File: matrix_mod_matrix_dmatrix.f90 ........................................................!:.:.:
!......................... type/class DMatrix procedures ..............................................................!:.:.:
!......................... Constructors ..............................................................................!:.:.:
 function           matrix_construct_zero_d                      (m,n,is,js,mtype,name)  result(MAT)                     !:.:.:
 function           matrix_construct_array2_d                    (C,is,js,mtype,name)    result(MAT)                     !:.:.:
```

```
 function           matrix_construct_real_d                        (r,m,n,is,js,mtype,name) result(MAT)                    !:.:.:
 function           matrix_construct_complex_d                     (r,m,n,is,js,mtype,name) result(MAT)                    !:.:.:
 function           matrix_construct_random_d                (rtype,m,n,is,js,mtype,name,sigma) result(MAT)                !:.:.:
 subroutine         matrix_symmetric_set_d                         (MAT,uplo)                                              !:.:.:
!......................... Operators    ..................................................................................!:.:.:
 pure subroutine    matrix_assignFrom_matrix_d                     (MATB,MATA)                                             !:.:.:
 subroutine         matrix_assignFrom_real_d                       (MATB,r)                                                !:.:.:
 subroutine         matrix_assignFrom_complex_d                    (MATB,r)                                                !:.:.:
 subroutine         matrix_assignFrom_array2_d                     (MATB,C)                                                !:.:.:
 pure function      real_plus_matrix_d                             (r,MATA)                result(MATB)                    !:.:.:
 pure function      matrix_plus_real_d                             (MATA,r)                result(MATB)                    !:.:.:
 function           matrix_plus_array2_d                           (MATA,C)                result(MATB)                    !:.:.:
 function           array2_plus_matrix_d                           (C,MATA)                result(MATB)                    !:.:.:
 function           matrix_plus_matrix_d                           (MATA,MATB)             result(MATC)                    !:.:.:
 pure function      real_subtract_matrix_d                         (r,MATA)                result(MATB)                    !:.:.:
 pure function      matrix_subtract_real_d                         (MATA,r)                result(MATB)                    !:.:.:
 function           matrix_subtract_array2_d                       (MATA,C)                result(MATB)                    !:.:.:
 function           array2_subtract_matrix_d                       (C,MATA)                result(MATB)                    !:.:.:
 function           matrix_subtract_matrix_d                       (MATA,MATB)             result(MATC)                    !:.:.:
 function           matrix_return_minus_matrix_d                   (MATA)                  result(MATB)                    !:.:.:
 pure function      real_mult_matrix_d                             (r,MATA)                result(MATB)                    !:.:.:
 pure function      matrix_mult_real_d                             (MATA,r)                result(MATB)                    !:.:.:
 pure function      complex_mult_matrix_d                          (z,MATA)                result(MATB)                    !:.:.:
 pure function      matrix_mult_complex_d                          (MATA,z)                result(MATB)                    !:.:.:
 function           matrix_mult_array2_d                           (MATA,C)                result(MATB)                    !:.:.:
 function           array2_mult_matrix_d                           (C,MATA)                result(MATB)                    !:.:.:
 function           matrix_mult_matrix_d                           (MATA,MATB)             result(MATC)                    !:.:.:
 pure function      matrix_divide_real_d                           (MATA,r)                result(MATB)                    !:.:.:
 pure subroutine    matrix_mult_matrix_sub_d                       (MATA,MATB,MATC)                                        !:.:.:
!......................... Linear Algebra with LAPACK ...............................................................!:.:.:
 function           dmatrix_inverse                                (MATA)                  result(MATB)                    !:.:.:
 function           dmatrix_determinant                            (MAT)                   result(z)                       !:.:.:
 function           dmatrix_lndet                                  (MAT)                   result(z)                       !:.:.:
 function           dmatrix_eigenvalues                            (MAT)                   result(vec)                     !:.:.:
 function           dmatrix_eigenvectors                           (MATA,vec)              result(MATB)                    !:.:.:
!......................... Misc Procedures...........................................................................!:.:.:
 subroutine         dmatrix_antisymmetric_set                      (MAT,uplo)                                              !:.:.:
 pure function      dmatrix_diagonal_get                           (MAT)                   result(vec)                     !:.:.:
 function           dmatrix_diagonal_set_from_dmatrix              (MATA)                  result(MATB)                    !:.:.:
 function           dmatrix_diagonal_set_from_dvector              (vec)                   result(MAT)                     !:.:.:
 pure function      dmatrix_traceless_get                          (MAT)                   result(MATB)                    !:.:.:
 pure subroutine    dmatrix_traceless_set                          (MAT)                                                   !:.:.:
 pure function      dmatrix_trace2c                                (MAT)                   result(r)                       !:.:.:
 pure function      dmatrix_trace2                                 (MAT)                   result(r)                       !:.:.:
 pure function      dmatrix_trace                                  (MAT)                   result(r)                       !:.:.:
 function           matrix_norm_d                                  (MAT)                   result(r)                       !:.:.:
 subroutine         random_number_dmatrix                          (MAT)                                                   !:.:.:
 subroutine         random_number_dmatrix_gaussian                 (MAT,sigma)                                             !:.:.:
 function           matrix_return_transpose_d                      (MATA)                  result(MATB)                    !:.:.:
 function           dmatrix_dmatrix_complex_return_matrix          (MATA,MATB)             result(MATC)                    !:.:.:
 function           dmatrix_is_symmetric                           (MATA)                  result(r)                       !:.:.:
 function           dmatrix_is_antisymmetric                       (MATA)                  result(r)                       !:.:.:
```

```
 function           dmatrix_is_nan                                (MAT)                    result(itis)                    !:.:.:
!........................ Math and Array Procedures ..................................................................!:.:.:
 pure function      dmatrix_abs                                   (MAT)                    result(MATB)                    !:.:.:
 pure function      dmatrix_sin                                    (MAT)                    result(MATB)                    !:.:.
 pure function      dmatrix_cos                                    (MAT)                    result(MATB)                    !:.:.
 pure function      dmatrix_exp                                    (MAT)                    result(MATB)                    !:.:.
 pure function      dmatrix_log                                    (MAT)                    result(MATB)                    !:.:.
 pure function      dmatrix_sqrt                                   (MAT)                    result(MATB)                    !:.:.
 pure function      dmatrix_power_integer                         (MAT,n)                  result(MATB)                    !:.:.:
 pure function      dmatrix_power_real                            (MAT,n)                  result(MATB)                    !:.:.:
!.....................................................................................................................!:.:.:
!........................ File: matrix_mod_matrix_vector.f90 ..........................................................!:.:.:
!........................ type/class Vector procedures ................................................................!:.:.:
!........................ Constructors ...............................................................................!:.:.:
 pure subroutine    vector_metadata_put                           (vec,n,is,name)                                          !:.:.:
 pure function      vector_construct_zero                         (n,is,name)              result(vec)                     !:.:.:
 pure function      vector_construct_real                         (n,r,is,name)            result(vec)                     !:.:.:
 pure function      vector_construct_complex                      (n,z,is,name)            result(vec)                     !:.:.:
 pure function      vector_construct_array1                       (C,is,name)              result(vec)                     !:.:.:
 function           vector_construct_random                       (rtype,n,is,name,sigma)  result(vec)                     !:.:.:
!........................ Procedures..................................................................................!:.:.:
 function           vector_sort                                   (veca,by)                result(vecb)                    !:.:.:
 pure function      vector_dot_product                            (veca,vecb)              result(r)                       !:.:.:
 subroutine         vector_random_set                             (vec)                                                    !:.:.:
 subroutine         vector_gaussian_set                           (vec,sigma)                                              !:.:.:
 subroutine         random_number_vector_gaussian_set             (vec,sigma)                                              !:.:.:
 pure subroutine    vector_metadata_copy_vector                   (veca,vecb)                                              !:.:.:
 pure subroutine    vector_metadata_copy_dvector                  (veca,vecb)                                              !:.:.:
 pure subroutine    dvector_metadata_copy_vector                  (veca,vecb)                                              !:.:.:
 pure subroutine    dvector_metadata_copy_dvector                 (veca,vecb)                                              !:.:.:
 pure function      vector_norm                                   (vec)                    result(r)                       !:.:.:
 pure function      vector_return_real_dvector                    (vec)                    result(vecb)                    !:.:.:
 pure function      vector_return_imag_dvector                    (vec)                    result(vecb)                    !:.:.:
 pure function      vector_return_return_conjg                    (vec)                    result(vecb)                    !:.:.:
!........................ Operators    ...............................................................................!:.:.:
 pure subroutine    vector_assignFrom_vector                      (vecb,veca)                                              !:.:.:
 pure subroutine    vector_assignFrom_real                        (vecb,r)                                                 !:.:.:
 pure subroutine    vector_assignFrom_complex                     (vecb,r)                                                 !:.:.:
 pure subroutine    vector_assignFrom_array1                      (vecb,v)                                                 !:.:.:
 pure subroutine    vector_assignFrom_array1_d                    (vecb,v)                                                 !:.:.:
 pure subroutine    vector_assignFrom_dvector                     (vecb,veca)                                              !:.:.:
 pure function      real_plus_vector                              (r,veca)                 result(vecb)                    !:.:.:
 pure function      vector_plus_real                              (veca,r)                 result(vecb)                    !:.:.:
 pure function      complex_plus_vector                           (z,veca)                 result(vecb)                    !:.:.:
 pure function      vector_plus_complex                           (veca,z)                 result(vecb)                    !:.:.:
 pure function      vector_plus_array1                            (veca,v)                 result(vecb)                    !:.:.:
 pure function      array1_plus_vector                            (v,veca)                 result(vecb)                    !:.:.:
 pure function      vector_plus_array1_d                          (veca,v)                 result(vecb)                    !:.:.:
 pure function      array1_d_plus_vector                          (v,veca)                 result(vecb)                    !:.:.:
 pure function      vector_plus_vector                            (veca,vecb)              result(vecc)                    !:.:.:
 pure function      vector_plus_dvector                           (veca,vecb)              result(vecc)                    !:.:.:
 pure function      dvector_plus_vector                           (veca,vecb)              result(vecc)                    !:.:.:
```

```
 pure function      real_subtract_vector                (r,veca)                result(vecb)                !:.:.:
 pure function      vector_subtract_real                (veca,r)                result(vecb)                !:.:.:
 pure function      complex_subtract_vector             (r,veca)                result(vecb)                !:.:.:
 pure function      vector_subtract_complex             (veca,r)                result(vecb)                !:.:.:
 pure function      vector_subtract_array1              (veca,C)                result(vecb)                !:.:.:
 pure function      array1_subtract_vector              (C,veca)                result(vecb)                !:.:.:
 pure function      vector_subtract_array1_d            (veca,C)                result(vecb)                !:.:.:
 pure function      array1_d_subtract_vector            (C,veca)                result(vecb)                !:.:.:
 pure function      vector_subtract_vector              (veca,vecb)             result(vecc)                !:.:.:
 pure function      vector_subtract_dvector             (veca,vecb)             result(vecc)                !:.:.:
 pure function      dvector_subtract_vector             (veca,vecb)             result(vecc)                !:.:.:
 pure function      vector_return_minus_vector          (veca)                  result(vecb)                !:.:.:
 pure function      real_mult_vector                    (r,veca)                result(vecb)                !:.:.:
 pure function      vector_mult_real                    (veca,r)                result(vecb)                !:.:.:
 pure function      complex_mult_vector                 (r,veca)                result(vecb)                !:.:.:
 pure function      vector_mult_complex                 (veca,r)                result(vecb)                !:.:.:
 function           matrix_mult_vector                  (MATA,vecb)             result(vecc)                !:.:.:
 function           vector_mult_matrix                  (vecb,MATA)             result(vecc)                !:.:.:
 pure function      vector_divide_real                  (veca,r)                result(vecb)                !:.:.:
 pure function      vector_divide_complex               (veca,r)                result(vecb)                !:.:.:
 pure subroutine    matrix_mult_vector_sub              (MATA,vecb,vecc)                                    !:.:.:
 pure subroutine    vector_mult_matrix_sub              (vecb,MATA,vecc)                                    !:.:.:
!......................... Math and Array Procedures ..................................................!:.:.:
 pure function      vector_abs                          (vec)                   result(vecb)                !:.:.:
 pure function      vector_sin                          (vec)                   result(vecb)                !:.:.:
 pure function      vector_cos                          (vec)                   result(vecb)                !:.:.:
 pure function      vector_exp                          (vec)                   result(vecb)                !:.:.:
 pure function      vector_log                          (vec)                   result(vecb)                !:.:.:
 pure function      vector_sqrt                         (vec)                   result(vecb)                !:.:.:
 pure function      vector_power_integer                (vec,n)                 result(vecb)                !:.:.:
 pure function      vector_power_real                   (vec,n)                 result(vecb)                !:.:.:
 pure function      vector_power_complex                (vec,n)                 result(vecb)                !:.:.:
!......................... Utilities ..................................................................!:.:.:
 function           vector_is_nan                       (vec)                   result(itis)                !:.:.:
 subroutine         vector_read                         (vec,unit)                                          !:.:.:
 subroutine         vector_save                         (vec,unit,fmt)                                      !:.:.:
 subroutine         vector_print                        (vec,unit,fmt,form,name,ips,ipe)                    !:.:.:
!.....................................................................................................!:.:.:
!......................... File: matrix_mod_matrix_dvector.f90 ........................................!:.:.:
!......................... type/class DVector procedures ..............................................!:.:.:
!......................... Constructors ...............................................................!:.:.:
 pure subroutine    dvector_metadata_put                (vec,n,is,name)                                     !:.:.:
 pure function      dvector_construct_zero              (n,is,name)             result(vec)                 !:.:.:
 pure function      dvector_construct_real              (n,r,is,name)           result(vec)                 !:.:.:
 pure function      dvector_construct_complex           (n,z,is,name)           result(vec)                 !:.:.:
 pure function      dvector_construct_array1            (C,is,name)             result(vec)                 !:.:.:
 function           dvector_construct_random            (rtype,n,is,name,sigma) result(vec)                 !:.:.:
!......................... Procedures..................................................................!:.:.:
 pure function      dvector_maxval                      (veca,mask)             result(r)                   !:.:.:
 pure function      dvector_minval                      (veca,mask)             result(r)                   !:.:.:
 pure function      dvector_dot_product                 (veca,vecb)             result(r)                   !:.:.:
 subroutine         dvector_random_set                  (vec)                                               !:.:.:
```

```
 subroutine         dvector_gaussian_set                        (vec,sigma)                                                !:.:.:
 subroutine         random_number_dvector_gaussian_set          (vec,sigma)                                                !:.:.:
 pure function      dvector_norm                                (vec)                  result(r)                           !:.:.:
 function           dvector_sort                                (veca,by)              result(vecb)                        !:.:.:
 function           dvector_dvector_complex_return_vector       (veca,vecb)            result(vecc)                        !:.:.:
!......................... Operators    ..................................................................................!:.:.:
 pure subroutine    dvector_assignFrom_vector                   (vecb,veca)                                                !:.:.:
 pure subroutine    dvector_assignFrom_real                     (vecb,r)                                                   !:.:.:
 pure subroutine    dvector_assignFrom_complex                  (vecb,r)                                                   !:.:.:
 pure subroutine    dvector_assignFrom_array1                   (vecb,v)                                                   !:.:.:
 pure subroutine    dvector_assignFrom_array1_d                 (vecb,v)                                                   !:.:.:
 pure function      real_plus_dvector                           (r,veca)               result(vecb)                        !:.:.:
 pure function      dvector_plus_real                           (veca,r)               result(vecb)                        !:.:.:
 pure function      dvector_plus_array1_d                       (veca,v)               result(vecb)                        !:.:.:
 pure function      array1_d_plus_dvector                       (v,veca)               result(vecb)                        !:.:.:
 pure function      dvector_plus_dvector                        (veca,vecb)            result(vecc)                        !:.:.:
 pure function      real_subtract_dvector                       (r,veca)               result(vecb)                        !:.:.:
 pure function      dvector_subtract_real                       (veca,r)               result(vecb)                        !:.:.:
 pure function      dvector_subtract_array1_d                   (veca,C)               result(vecb)                        !:.:.:
 pure function      array1_d_subtract_dvector                   (C,veca)               result(vecb)                        !:.:.:
 pure function      dvector_subtract_dvector                    (veca,vecb)            result(vecc)                        !:.:.:
 pure function      dvector_return_minus_dvector                (veca)                 result(vecb)                        !:.:.:
 pure function      real_mult_dvector                           (r,veca)               result(vecb)                        !:.:.:
 pure function      dvector_mult_real                           (veca,r)               result(vecb)                        !:.:.:
 function           dmatrix_mult_dvector                        (MATA,vecb)            result(vecc)                        !:.:.:
 function           dvector_mult_dmatrix                        (vecb,MATA)            result(vecc)                        !:.:.:
 pure function      dvector_divide_real                         (veca,r)               result(vecb)                        !:.:.:
 pure subroutine    matrix_mult_vector_sub_d                   (MATA,vecb,vecc)                                            !:.:.:
 pure subroutine    vector_mult_matrix_sub_d                   (vecb,MATA,vecc)                                            !:.:.:
!......................... Math and Array Procedures ..........................................................................!:.:.:
 pure function      dvector_abs                                 (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_sin                                 (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_cos                                 (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_exp                                 (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_log                                 (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_sqrt                                (vec)                  result(vecb)                        !:.:.:
 pure function      dvector_power_integer                       (vec,n)                result(vecb)                        !:.:.:
 pure function      dvector_power_real                          (vec,n)                result(vecb)                        !:.:.:
!......................... Utilities ..........................................................................................!:.:.:
 function           dvector_is_nan                              (vec)                  result(itis)                        !:.:.:
 subroutine         dvector_read                                (vec,unit)                                                 !:.:.:
 subroutine         dvector_save                                (vec,unit,fmt)                                             !:.:.:
 subroutine         dvector_print                               (vec,unit,fmt,form,name,ips,ipe)                           !:.:.:
!..............................................................................................................................!:.:.:
!......................... File: matrix_mod_array_lapack.f90 ..................................................................!:.:.:
!..............................................................................................................................!:.:.:
!......................... Matrix Matrix Multiplication .......................................................................!:.:.:
 pure function      array2_matmul_lapack_mm                     (A,B,mtype)            result(C)                           !:.:.:
   call zhemm(side,uplo,mc,nc   ,alpha,A,ma,B,mb,beta,C,mc)  ! A is assumed Hermitian                                      !:.:.:
   call zgemm(opA ,opB ,mc,nc,na,alpha,A,ma,B,mb,beta,C,mc)                                                               !:.:.:
 pure function      array2_matmul_lapack_dd                     (A,B,mtype)            result(C)                           !:.:.:
   call dsymm(side,uplo,mc,nc   ,alpha,A,ma,B,mb,beta,C,mc)  ! A is assumed Symmetric                                     !:.:.:
```

```
   call dgemm(opA ,opB ,mc,nc,na,alpha,A,ma,B,mb,beta,C,mc)                                                    !:.:.:
 pure function      array2_matmul_lapack_m                          (A,side)              result(C)           !:.:.:
  call zherk(uplo, opA ,N,K,alpha,A,ma,beta,C,N)                                                             !:.:.:
 pure function      array2_matmul_lapack_d                          (A,side)              result(C)           !:.:.:
  call dsyrk(uplo, opA ,N,K,alpha,A,ma,beta,C,N)                                                             !:.:.:
!.......................... Matrix Vector Multiplication ..................................................!:.:.:
 pure function      array2_matmul_lapack_mv                         (A,v,type)            result(w)           !:.:.:
    call zgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                       !:.:.:
    call zhemv(uplo ,ma,alpha,A,ma,v,incx,beta,w,incy)                                                       !:.:.:
 pure function      array2_matmul_lapack_dv                         (A,v,type)            result(w)           !:.:.:
  call dgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                         !:.:.:
!.......................... Matrix Matrix Multiplication Subroutines ......................................!:.:.:
 pure subroutine    array2_matmul_lapack_mm_sub                     (A,B,C,mtype)                           !:.:.:
   call zhemm(side,uplo,mc,nc   ,alpha,A,ma,B,mb,beta,C,mc)  ! A is assumed Hermitian                       !:.:.:
   call zgemm(opA ,opB ,mc,nc,na,alpha,A,ma,B,mb,beta,C,mc)                                                 !:.:.:
 pure subroutine    array2_matmul_lapack_dd_sub                     (A,B,C,mtype)                           !:.:.:
   call dsymm(side,uplo,mc,nc   ,alpha,A,ma,B,mb,beta,C,mc)  ! A is assumed Symmetric                       !:.:.:
   call dgemm(opA ,opB ,mc,nc,na,alpha,A,ma,B,mb,beta,C,mc)                                                 !:.:.:
 pure subroutine    array2_matmul_lapack_m_sub                      (A,C,side)                              !:.:.:
  call zherk(uplo, opA ,N,K,alpha,A,ma,beta,C,N)                                                           !:.:.:
 pure subroutine    array2_matmul_lapack_d_sub                      (A,C,side)                              !:.:.:
  call dsyrk(uplo, opA ,N,K,alpha,A,ma,beta,C,N)                                                           !:.:.:
!.......................... Matrix Vector Multiplication Subroutines ......................................!:.:.:
pure subroutine     array2_matmul_lapack_mv_sub                     (A,v,w,type)                            !:.:.:
    call zgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                     !:.:.:
    call zhemv(uplo ,ma,alpha,A,ma,v,incx,beta,w,incy)                                                     !:.:.:
pure subroutine     array2_matmul_lapack_vm_sub                     (v,A,w)                                 !:.:.:
  call zgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                       !:.:.:
pure subroutine     array2_matmul_lapack_dv_sub                     (A,v,w,type)                            !:.:.:
  call dgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                       !:.:.:
pure subroutine     array2_matmul_lapack_vd_sub                     (v,A,w)                                 !:.:.:
  call dgemv(tp,ma,na,alpha,A,ma,v,incx,beta,w,incy)                                                       !:.:.:
!.......................... Matrix Inversion ..............................................................!:.:.:
 function           array2_inverse                                  (C)                   result(CI)          !:.:.:
  call zgetrf(n,n,CI,n,ipiv,info)                                                                          !:.:.:
  call zgetri(n,CI,n,ipiv,WORK    ,LWORK,info)                                                             !:.:.:
 function           array2_inverse_d                                (C)                   result(CI)          !:.:.:
  call dgetrf(n,n,CI,n,ipiv,info)                                                                          !:.:.:
  call dgetri(n,CI,n,ipiv,WORK    ,LWORK,info)                                                             !:.:.:
!.......................... Eigenvalues - Eigenvectors ....................................................!:.:.:
 function           array2_eigenvalues                              (C,mtype)             result(eigenval)    !:.:.:
 function           array2_eigenvectors                             (C,mtype)             result(evs)         !:.:.:
 function           array2_eigenvalues_d                            (C,mtype)             result(eigenval)    !:.:.:
 function           array2_eigenvectors_d                           (C,mtype)             result(evs)         !:.:.:
 subroutine         array2_zgeev                                    (C,eigenval,eigenvec,job)               !:.:.:
   call zgeev(JOBVL,JOBVR,n,A,n,EV,VL,LDVL,VR,LDVL,WORK    ,LWORK,RWORK,info)                                !:.:.:
 subroutine         array2_zheev                                    (C,eigenval,eigenvec,job)               !:.:.:
   call zheev(JOBZ,UPLO,n,A,n,EV,WORK    ,LWORK,RWORK,info)                                                !:.:.:
 subroutine         array2_dgeev                                    (C,eigenval,eigenvec,job)               !:.:.:
   call dgeev(JOBVL,JOBVR,n,A,n,ReEV,ImEV,VL,LDVL,VR,LDVL,WORK    ,LWORK,info)                              !:.:.:
 subroutine         array2_dsyev                                    (C,eigenval,eigenvec,job)               !:.:.:
   call dsyev(JOBZ,UPLO,n,A,n,EV,WORK    ,LWORK,info)                                                      !:.:.:
```

```
 function           array2_determinant                          (C)  result (det)                                  !:.:.:
  call zgetrf(n,n,A,n,ipiv,info)                                                                                   !:.:.:
 function           array2_log_determinant                      (C)                    result(det)                 !:.:.:
  call zgetrf(n,n,A,n,ipiv,info)                                                                                   !:.:.:
 function           array2_determinant_d                        (C)                    result(det)                 !:.:.:
  call dgetrf(n,n,A,n,ipiv,info)                                                                                   !:.:.:
 function           array2_log_determinant_d                    (C)                    result(det)                 !:.:.:
  call dgetrf(n,n,A,n,ipiv,info)                                                                                   !:.:.:
 function                                                       array2_pfaffian(C)     result(pfaffian)            !:.:.:
 function           array2_log_pfaffian                         (C)                    result(pfaffian)            !:.:.:
 function           array2_pfaffian2                            (C)                    result(pfaffian)            !:.:.:
 function           array2_log_pfaffian2                        (C)                    result(pfaffian)            !:.:.:
!.....................................................................................................................!:.:.:
!.....................................................................................................................!:.:.:
!..................... File: tensorprod_mod.f90   ....................................................................!:.:.:
module              tensorprod_mod                                                                                    !:
!---------------------------------------------------------------------------------------------------------------------!:.:.:
 public                                :: tensorprod, vec2row, vec2col, vectorize                                    !:.:.:
!---------------------------------------------------------------------------------------------------------------------!:.:.:
 interface          tensorprod                                                                                       !:.:.:
 pure function      tensorprod_complex_complex                  (X,Y)                  result(Z)                   !:.:.:
 pure function      tensorprod_complex_3                        (X1,X2,X3)             result(Z)                   !:.:.:
 pure function      tensorprod_complex_4                        (X1,X2,X3,X4)          result(Z)                   !:.:.:
 pure function      tensorprod_complex_5                        (X1,X2,X3,X4,X5)       result(Z)                   !:.:.:
 pure function      tensorprod_complex_6                        (X1,X2,X3,X4,X5,X6)    result(Z)                   !:.:.:
 pure function      tensorprod_real_real                        (X,Y)                  result(Z)                   !:.:.:
 pure function      tensorprod_real_3                           (X1,X2,X3)             result(Z)                   !:.:.:
 pure function      tensorprod_real_4                           (X1,X2,X3,X4)          result(Z)                   !:.:.:
 pure function      tensorprod_real_5                           (X1,X2,X3,X4,X5)       result(Z)                   !:.:.:
 pure function      tensorprod_real_6                           (X1,X2,X3,X4,X5,X6)    result(Z)                   !:.:.:
 pure function      tensorprod_complex_vec_vec                  (u,v)                  result(Z)                   !:.:.:
 pure function      tensorprod_real_vec_vec                     (u,v)                  result(Z)                   !:.:.:
 pure function      vec2col_complex                             (u)                    result(Z)                   !:.:.:
 pure function      vec2row_complex                             (u)                    result(Z)                   !:.:.:
 pure function      vec2col_real                                (u)                    result(Z)                   !:.:.:
 pure function      vec2row_real                                (u)                    result(Z)                   !:.:.:
 pure function      vectorize_complex                           (Z)                    result(u)                   !:.:.:
 pure function      vectorize_real                              (Z)                    result(u)                   !:.:.:
```