Fortran 95
intrinsic

Uwe Kuester

• taken from

The Queen's University of Belfast
Parallel Computer Centre
http://www.pcc.qub.ac.uk/tec/courses/f90/stu-notes/F90_notesMIF_13.html
• arguments are named according to their implicit types
• Optional arguments in square brackets [],
• keywords for the argument names are those given.
  – KIND
    • describes the KIND number.
  – SET
    • a string containing a set of characters.
  – BACK
    • a logical used to determine the direction a string is to be searched.
  – MASK
    • a logical array used to identify those element which are to take part in the desired operation.
  – DIM
    • a selected dimension of an argument (an integer).

functions and subroutines
• nearly all intrinsics are functions
• reflects the significance of the function results in connection with array syntax and (not for intrinsics) derived types
1 Argument presence enquiry

- \text{PRESENT}(A)
  - true if A is present.

2 Numeric functions 1

- \text{ABS}(A)
  - return the absolute value of A.
- \text{AIMAG}(Z)
  - return the imaginary part of complex number Z.
- \text{AINT}(A, \text{KIND})
  - returns a value A truncated to a whole number.
- \text{ANINT}(A, \text{KIND})
  - returns a value rounded to the nearest value of A.
- \text{CEILING}(A)
  - returns the lowest integer greater than or equal to A.
- \text{CMPLX}(X, Y, \text{KIND})
  - converts A to a complex number.
- \text{CONJG}(Z)
  - returns the conjugate of a complex number.
- \text{DBLE}(A)
  - converts A to a double precision real.
2 Numeric functions 2

- **DIM( X, Y )**
  - returns the maximum of X-Y or 0.
- **DPROD( X, Y )**
  - returns a double precision product.
- **FLOOR( A )**
  - returns the largest integer less than or equal to A.
- **INT( A [, KIND] )**
  - converts to an integer.
- **MAX( A1, A2 [, A3...] )**
  - returns the maximum value.
- **MIN( A1, A2 [, A3...] )**
  - returns the minimum value.
- **MOD( A, P )**
  - returns remainder modulo P i.e. A-INT(A/P)*P.
- **MODULO( A, P )**
  - A modulo P.

2 Numeric functions 3

- **NINT( A [, KIND] )**
  - returns the nearest integer to A.
- **REAL( A [, KIND] )**
  - converts to a real.
- **SIGN( A, B )**
  - returns the absolute value of A times the sign of B.
3 Mathematical functions 1

- **ACOS( X )**
  - arccosine.
- **ASIN( X )**
  - arcsine.
- **ATAN( X )**
  - arctan.
- **ATAN2( X, Y )**
  - arctan.
- **COS( X )**
  - cosine.
- **COSH( X )**
  - hyperbolic cosine.
- **EXP( X )**
  - exponential.
- **LOG( X )**
  - natural logarithm.

3 Mathematical functions 2

- **LOG10( X )**
  - base 10 logarithm.
- **SIN( X )**
  - sine.
- **SINH( X )**
  - hyperbolic sine.
- **SQRT( X )**
  - square root.
- **TAN( X )**
  - tan.
- **TANH( X )**
  - hyperbolic tan.
4 Character functions 1

- **ACHAR( I )**
  - returns the Ith character in the ASCII collating sequence.
- **ADJUSTL( STRING )**
  - adjusts string left by removing any leading blanks and inserting trailing blanks.
- **ADJUSTR( STRING )**
  - adjusts string right by removing trailing blanks and inserting leading blanks.
- **CHAR( I [, KIND] )**
  - returns the Ith character in the machine specific collating sequence.
- **IACHAR( C )**
  - returns the position of the character in the ASCII collating sequence.
- **ICHAR( C )**
  - returns the position of the character in the machine specific collating sequence.
- **INDEX( STRING, SUBSTRING [, BACK] )**
  - returns the leftmost (rightmost if BACK is .TRUE.) starting position of SUBSTRING within STRING.

4 Character functions 2

- **LEN( STRING )**
  - returns the length of a string.
- **LEN_TRIM( STRING )**
  - returns the length of a string without trailing blanks.
- **LGE( STRING_A, STRING_B )**
  - lexically greater than or equal to.
- **LGT( STRING_A1, STRING_B )**
  - lexically greater than.
- **LLE( STRING_A, STRING_B )**
  - lexically less than or equal to.
- **LLT( STRING_A, STRING_B )**
  - lexically less than.
- **REPEAT( STRING, NCOPIES )**
  - repeats concatenation.
- **SCAN( STRING, SET [, BACK] )**
  - returns the index of the leftmost (rightmost if BACK is .TRUE.) character of STRING that belong to SET, or 0 if none belong.
4 Character functions 3

- TRIM( STRING )
  - removes trailing spaces from a string.
- VERIFY( STRING, SET [, BACK] )
  - returns zero if all characters in STRING belong to SET or the index of the leftmost (rightmost if BACK is .TRUE.) that does not.

5 KIND functions

- KIND( X )
  - returns the kind type parameter value.
- SELECTED_INT_KIND( R )
  - kind of type parameter for specified exponent range.
- SELECTED_REAL_KIND( [P] [,R] )
  - kind of type parameter for specified precision and exponent range.
6 Logical functions

- LOGICAL(L [, KIND])
  - convert between different logical kinds.

7 Numeric enquiry functions 1

- DIGITS(X)
  - returns the number of significant digits in the model.
- EPSILON(X)
  - returns the smallest value such that REAL(1.0, KIND(X)) + EPSILON(X) is not equal to REAL(1.0, KIND(X)).
- HUGE(X)
  - returns the largest number in the model.
- MAXEXPONENT(X)
  - returns the maximum exponent value in the model.
- MINEXPONENT(X)
  - returns the minimum exponent value in the model.
- PRECISION(X)
  - returns the decimal precision.
- RADIX(X)
  - returns the base of the model.
7 Numeric enquiry functions 2

- RANGE( X )
  - returns the decimal exponent range.
- TINY( X )
  - returns the smallest positive number in the model.

8 Bit enquiry functions

- BIT_SIZE( I )
  - returns the number of bits in the model.
9 Bit manipulation functions 1

- BTEST(I, POS)
  - is .TRUE. if bit POS of integer I has a value 1.
- IAND(I, J)
  - logical .AND. on the bits of integers I and J.
- IBCLR(I, POS)
  - clears bit POS of integer I to 0.
- IBITS(I, POS, LEN)
  - extracts a sequence of bits length LEN from integer I starting at POS.
- IBSET(I, POS)
  - sets bit POS of integer I to 1.
- IEOR(I, J)
  - performs an exclusive .OR. on the bits of integers I and J.
- IOR(I, J)
  - performs an inclusive .OR. on the bits of integers I and J.
- ISHIFT(I, SHIFT)
  - logical shift of the bits.

9 Bit manipulation functions 2

- ISHIFTC(I, SHIFT [, SIZE])
  - logical circular shift on a set of bits on the right.
- NOT(I)
  - logical complement on the bits.
10 Transfer functions

- \texttt{TRANSFER( SOURCE, MOLD [, SIZE] )}
  - converts \texttt{SOURCE} to the type of \texttt{MOLD}.

11 Floating point manipulation functions

- \texttt{EXPONENT( X )}
  - returns the exponent part of \texttt{X}.
- \texttt{FRACTION( X )}
  - returns the fractional part of \texttt{X}.
- \texttt{NEAREST( X, S )}
  - returns the nearest different machine specific number in the direction given by the sign of \texttt{S}.
- \texttt{RRSPACING( X )}
  - returns the reciprocal of the relative spacing of model numbers near \texttt{X}.
- \texttt{SCALE( X )}
  - multiple \texttt{X} by its base to power \texttt{I}.
- \texttt{SET_EXPONENT( X, I )}
  - sets the exponent part of \texttt{X} to be \texttt{I}.
- \texttt{SPACING( X )}
  - returns the absolute spacing of model numbers near \texttt{X}.
12 Vector and matrix functions

- DOT_PRODUCT(VECTOR_A, VECTOR_B)
  - returns the dot product of two vectors (rank one arrays).

- MATMUL(MATRIX_A, MATRIX_B)
  - returns the product of two matrices.

13 Array reduction functions

- ALL(MASK [, DIM])
  - returns .TRUE. if all elements of MASK are .TRUE.

- ANY(MASK [, DIM])
  - returns .TRUE. if any elements of MASK are .TRUE.

- COUNT(MASK [, DIM])
  - returns the number of elements of MASK that are .TRUE.

- MAXVAL(ARRAY [, DIM] [, MASK])
  - returns the value of the maximum array element.

- MINVAL(ARRAY [, DIM] [, MASK])
  - returns the value of the minimum array element.

- PRODUCT(ARRAY [, DIM] [, MASK])
  - returns the product of array elements

- SUM(ARRAY [, DIM] [, MASK])
  - returns the sum of array elements.
14 Array enquiry functions

- **ALLOCATED( ARRAY )**
  - returns .TRUE. if ARRAY is allocated.
- **LBOUND( ARRAY [, DIM] )**
  - returns the lower bounds of the array.
- **SHAPE( SOURCE )**
  - returns the array (or scalar) shape.
- **SIZE( ARRAY [, DIM] )**
  - returns the total number of elements in an array.
- **UBOUND( ARRAY [, DIM] )**
  - returns the upper bounds of the array.

15 Array constructor functions

- **MERGE( TSOURCE, FSOURCE, MASK )**
  - returns value(s) of TSOURCE when MASK is .TRUE. and FSOURCE otherwise.
- **PACK( ARRAY, MASK [, VECTOR] )**
  - pack elements of ARRAY corresponding to true elements of MASK into a rank one result
- **SPREAD( SOURCE, DIM, NCOPIES )**
  - returns an array of rank one greater than SOURCE containing NCOPIES of SOURCE.
- **UNPACK( VECTOR, MASK, FIELD )**
  - unpack elements of VECTOR corresponding to true elements of MASK.
16 Array reshape and manipulation functions

- CSHIFT( ARRAY, SHIFT [, DIM] )
  - performs a circular shift.
- EOSHIFT( ARRAY, SHIFT [, BOUNDARY] [, DIM] )
  - performs an end-off shift.
- MAXLOC( ARRAY [, MASK] )
  - returns the location of the maximum element.
- MINLOC( ARRAY [, MASK] )
  - returns the location of the minimum element.
- RESHAPE( SOURCE, SHAPE [, PAD] [, ORDER] )
  - rehapes SOURCE to shape SHAPE
- TRANSPOSE( MATRIX )
  - transpose a matrix (rank two array).

17 Pointer association status enquiry functions

- ASSOCIATED( POINTER [, TARGET] )
  - returns .TRUE. if POINTER is associated.
18 Intrinsic subroutines

- CPU_TIME(TIME)
  - CPU time in seconds
- DATE_AND_TIME([DATE] [, TIME] [, ZONE] [, VALUES])
  - real time clock reading date and time.
- MVBITS( FROM, FROMPOS, LEN, TO TOPOS)
  - copy bits.
- RANDOM_NUMBER( HARVEST )
  - random number in the range 0-1 (inclusive).
- RANDOM_SEED([SIZE] [, PUT] [, GET])
  - initialise or reset the random number generator.
- SYSTEM_CLOCK([COUNT] [, COUNT_RATE] [, COUNT_MAX])
  - integer data from the real time clock.