Table A.47: F90 Selective Single Inheritance Form

```
module derived_class_name
      use base_class_name, only: list_of_entities
! new attribute declarations, if any
                  ...
contains
    ! new member definitions
                  ...
end module derived_class_name
```

Table A.48: F90 Single Inheritance Form with Local Renaming

```
module derived_class_name
      use base_class_name, local_name => base_entity_name
! new attribute declarations, if any
                  ...
contains
    ! new member definitions
                  ...
end module derived_class_name
```

Table A.49: F90 Multiple Selective Inheritance with Renaming

```
module derived_class_name
      use base1_class_name
      use base2_class_name
      use base3_class_name, only: list_of_entities
      use base4_class_name, local_name => base_entity_name
! new attribute declarations, if any
                  ...
contains
      ! new member definitions
                  ...
end module derived_class_name
```

A.2 Alphabetical Table of Fortran 90 Intrinsic Routines

The following KEY symbols are utilized to denote the TYPE of the intrinsic function, or subroutine, and its arguments: A-complex, integer, or real; B-integer bit; C-character; D-dimension; I-integer; K-kind; L-logical; M-mask (logical); N-integer, or real; P-pointer; R-real; S-string; T-target; V-vector (rank-one array); X-real; Y-real; Z-complex; and *-any type. For more detailed descriptions and sample uses of these intrinsic functions, see [1].

Alphabetical Table of Fortran 90 Intrinsic Functions

Type	Intrinsic	Description
A	ABS (A)	Absolute value of A
C	ACHAR (I)	Character in position I of ASCII collating sequence
R	ACOS (X)	Arc cosine (inverse cosine) function of real X
C	ADJUSTL (S)	Adjust S left; move leading blanks to trailing blanks
C	ADJUSTR (S)	Adjust S right; move trailing blanks to leading blanks
R	AIMAG (Z)	Imaginary part of complex number Z
R	AINT (X [,K])	Truncate X to a real whole number of the given kind
L	ALL (M [,D])	True if all mask M elements are true in dimension D
L	ALLOCATED (*_ARRAY_P)	True if the array or pointer is allocated
R	ANINT (X [,K])	Real whole number nearest to X of the given kind
L	ANY (M [,D])	True if any mask M element is true in dimension D
R	ASIN (X)	Arcsine (inverse sine) function of real X
L	ASSOCIATED (P [,T])	True if pointer P is associated with any target or T
R	ATAN (X)	Arctangent (inverse tangent) function of real X
R	ATAN2 (Y,X)	Arctangent for argument of complex number (X, Y)
I	BIT_SIZE (I)	Maximum number of bits integer I can hold (e.g., 32)
L	BTEST (I,I_POS)	True if bit location I_POS of integer I has value 1
I	CEILING (X)	Least integer ≥ real X of the given kind
C	CHAR (I [,K])	Character in position I of processor collating sequence
Z	CMPLX (X [,Y][,K])	Convert real(s) to complex type of given kind
Z	CONJG (Z)	Conjugate of complex number Z
R	COS (R_Z)	Cosine function of real or complex argument
R	COSH (X)	Hyperbolic cosine function of real X
I	COUNT (M [,D])	Number of true mask M elements in dimension D
*	CSHIFT (*_ARAY,I_SHIF [,D])	Circular shift out and in for I_SHIF elements
call	DATE_AND_TIME ([S_DATE] [,S_TIME] [,S_ZONE] [,I_V_VALUES])	Real-time clock date, time, zone, and vector with year, month, day, UTC, hour, minutes, seconds, and milliseconds
R	DBLE (A)	Convert A to double-precision real
N	DIGITS (N)	Number of significant digits for N (e.g., 31)
R	DIM (X,Y)	The difference, MAX (X – Y, 0.0)
N,L	DOT_PRODUCT (V,V_2)	Dot product of vectors V and V_2
R	DPROD (X,Y)	Double-precision real product of two real scalars
*	EOSHIFT (*_ARRAY, I_SHIFT [,*_FILL][,D])	Perform vector end-off shift by ± I_shift terms and fill in dimension D
R	EPSILON (X)	Number ≪ 1 for numbers like X (e.g. 2**–23)
R,Z	EXP (R_Z)	Exponential function of real or complex argument
I	EXPONENT (X)	Exponent part of the model for real X
I	FLOOR (X)	Greatest integer less than or equal to X
R	FRACTION (X)	Fractional part of the model for real X
N	HUGE (N)	Largest number for numbers like N (e.g., 2**128)
I	IACHAR (C)	Position of character C in ASCII collation
B	IAND (I,I_2)	Logical AND on the bits of I and I_2
B	IBCLR (I,I_POS)	Clear bit I_POS to zero in integer I
B	IBITS (I,I_POS,I_LEN)	Extract an I_LEN sequence of bits at I_POS in I
B	IBSET (I,I_POS)	Set bit I_POS to one in integer I
I	ICHAR (C)	Position of character C in processor collation
B	IEOR (I,I_2)	Exclusive OR on the bits of I and I_2
I	INDEX (S,S_SUB [,L_BACK])	Left starting position of S_SUB within S (right)

(continued)

(continued)

Type	Intrinsic	Description
I	INT (A [,K])	Convert A to integer type of given kind
B	IOR (I,I_2)	Inclusive OR on the bits of I and I_2
B	ISHFT (I,I_SHIFT)	Logical shift of bits of I by I_SHIFT pad with 0
B	ISHFTC (I,I_SHIFT [,I_SIZE])	Logical circular shift of I_SIZE rightmost bits of I
I	KIND (ANY)	Kind type integer parameter value for any argument
I,V	LBOUND (*_ARRAY [,D])	ARRAY lower bound(s) vector along dimension D
I	LEN (S)	Total character string length
I	LEN_TRIM (S)	Length of S without trailing blanks
L	LGE (S,S_2)	True if S > or equal to S_2 in ASCII sequence
L	LGT (S,S_2)	True if S follows S_2 in ASCII collating sequence
L	LLE (S,S_2)	True if S < or equal to S_2 in ASCII sequence
L	LLT (S,S_2	True if S precedes S_2 in ASCII collating sequence
R	LOG (R_Z)	Natural (base e) logarithm of real or complex number
L	LOGICAL (L [,K])	Convert L to logical of kind K
R	LOG10 (X)	Common (base 10) logarithm function of real X
N,L	MATMUL (MATRIX,MATRIX_2)	Conformable matrix multiplication
N	MAX (N,N_2 [,N_3,...])	Maximum value of two or more numbers of the same type
I	MAXEXPONENT (X)	Maximum exponent for real numbers like X (e.g. 128)
I,V	MAXLOC (N_ARRAY [,M])	Location(s) of maximum ARRAY element passing M
N	MAXVAL (N_ARRAY [,D] [,M])	Maximum ARRAY term in dimension D passing M
*	MERGE (*_TRUE,*_FALSE,M)	Use *_TRUE when M is true; *_FALSE otherwise
N	MIN (N,N_2 [,N_3,...])	Minimum value of two or more same type numbers
I	MINEXPONENT (X)	Minimum exponent for real numbers like X (e.g., –125)
I,V	MINLOC (N_ARRAY [,M])	Location(s) of minimum ARRAY term, passing M
N	MINVAL (N_ARRAY [,D] [,M])	Minimum ARRAY term in dimension D passing M
N	MOD (N,N_2)	Remainder for N_2. That is, N–INT(N/N_2)* N_2
N	MODULO (N,N_2)	Modulo, that is, N–FLOOR(N/N_2)*N_2
call	MVBITS (I_FROM,I_LOC, I_LEN,I_TO,I_POS)	Copy I_LEN bits at I_LOC in I_FROM to I_TO at I_POS
R	NEAREST (X,Y)	Nearest number at X in the direction of sign Y
I	NINT (X [,K])	Integer nearest to real X of the stated kind
I	NOT (I)	Logical complement of the bits of integer I
*,V	PACK (*_ARRAY,M [,V_PAD])	Pack ARRAY at true M into vector using V_PAD
I	PRECISION (R_Z)	Decimal precision for a real or complex R_Z (e.g., 6)
L	PRESENT (OPTIONAL)	True if optional argument is present in call
A	PRODUCT (A_ARRAY [,D] [,M])	Product of ARRAY elements along D for mask M
I	RADIX (N)	Base of the model for numbers like N (e.g., 2)
call	RANDOM_NUMBER (X)	Pseudorandom numbers in range $0 < X < 1$
call	RANDOM_SEED ([I_SIZE] [,I_V_PUT][,I_V_GET])	Initialize random number generator; defaults to processor initialization
I	RANGE (A)	Decimal exponent range in the model for A (e.g., 37)
R	REAL (A [,K])	Convert A to real type of type K
S	REPEAT (S,I_COPIES)	Concatenates I_COPIES of string S
*	RESHAPE (*_ARAY,I_V_SHAP [,*_PAD] [,V_ORDER])	Reshape ARAY using vector SHAP pad from an array and reorder
R	RRSPACING (X)	Relative spacing reciprocal of numbers near X
R	SCALE (X,I)	Return X times b**I, for base of b = RADIX (X)
I	SCAN (S,S_SET [,L_BACK])	Leftmost character index in S found in S_SET; (rightmost)
I	SELECTED_INT_KIND (I_r)	Integer kind with range, –(10**I_r) to (10**I_r)

Type	Intrinsic	Description
I	SELECTED_REAL_KIND	Kind for real of decimal precision I and exponent
I	([I] [,I_r])	range I_r
R	SET_EXPONENT (X,I)	Number with mantissa of X and exponent of I
I,V	SHAPE (*_ARRAY)	ARRAY (or scalar) shape vector
N	SIGN (N,N_2)	Absolute value of N times sign of same type N_2
R,Z	SIN (R_Z)	Sine function of real or complex number
R	SINH (X)	Hyperbolic sine function of real X
I	SIZE (*_ARRAY [,D])	ARRAY size along dimension D
R	SPACING (X)	Absolute spacing of numbers near real X (e.g., 2**–17)
*	SPREAD (*_ARAY,D,I_COPIES)	I_COPIES along dimension D of ARAY into an array of rank 1 greater
R,Z	SQRT (R_Z)	Square root function of real or complex number
A	SUM (A_ARRAY [,D] [,M])	Sum of ARRAY elements along D passing mask M
call	SYSTEM_CLOCK ([I_NOW]	Integer data from real-time clock. CPU time is
	[,I_RATE] [,I_MAX])	(finish_now - start_now) / rate
R	TAN (X)	Tangent function of real X
R	TANH (X)	Hyperbolic tangent function of real X
R	TINY (N)	Smallest positive number, like N (e.g., 2**–126)
*	TRANSFER (*_ARAY, V_MOLD	Same representation as ARAY but type of MOLD in
	[,I_SIZE])	vector of length SIZE
*	TRANSPOSE (MATRIX)	Matrix transpose of any type matrix
S	TRIM (S)	Remove trailing blanks from a single string
I,V	UBOUND (*_ARRAY [,D])	ARRAY upper bound(s) vector along dimension D
*	UNPACK (V,M,*_USE)	Unpack vector V at true elements of M into USE
I	VERIFY (S,S_SET [,L_BACK])	First position in S not found in S_SET (or last)

A.3 Subject Table of Fortran 90 Intrinsic Routines

The following KEY symbols are utilized to denote the TYPE of the intrinsic function, or subroutine, and its arguments: A-complex, integer, or real; B-integer bit; C-character; D-dimension; I-integer; K-kind; L-logical; M-mask (logical); N-integer, or real; P-pointer; R-real; S-string; T-target; V-vector (rank one array); X-real; Y-real; Z-complex; and *-any type. For more detailed descriptions and illustrative uses of these intrinsic functions, see Adams, J.C. et al., [1].

Subject Table of Fortran 90 Intrinsic Routines

Type	Intrinsic	Description
ALLOCATION		
L	ALLOCATED (*_ARRAY)	True if the array is allocated
ARGUMENT		
L	PRESENT (OPTIONAL)	True if optional argument is present in the call
ARRAY: CONSTRUCTION		
*	MERGE (*_TRUE,*_FALSE,M)	Use *_TRUE if M is true; *_FALSE otherwise
*,V	PACK (*_ARRAY,M [,V_PAD])	Pack ARRAY for true M into vector and pad from V_PAD
*	RESHAPE (*_ARRAY,I_V_SHAPE [,*_PAD] [,V_ORDER])	Reshape ARRAY using vector SHAPE, pad from an array, and reorder

(continued)