

X3J3/96-131r1

From: Loren P. Meissner

Subject: Extend Character Intrinsic

Meeting 138

<FORTREQ>

<NUMBER>

<TITLE> Extend Max, Min, etc. to Character Data Type

<KEYWORDS> Intrinsic functions, Character type

<STATUS> Registered

<TARGET>

<SUBGROUP>

<VERSION> 3

<REQUIREMENT>

Extend the intrinsic functions Max, Min, MaxLoc, MinLoc, MaxVal, and MinVal to accept arguments of character data type.

</REQUIREMENT>

<JUSTIFICATION>

Regularity is the primary reason for this requirement. These intrinsic functions now accept arguments of the other two ordered intrinsic types, integer and real. This proposal recognizes character data type as a first-class type and extends the ordering intrinsics so that they accept all types for which they make sense. This extension will facilitate construction of families of similar routines in applications such as sorting for integer, real, and character data.

</JUSTIFICATION>

<SUGGESTED IMPLEMENTATION>

Add text to the specifications of the intrinsic functions mentioned. Intrinsic functions for ordering character arguments should be consistent with character relational operators: for example, when A and B are character strings Max( A, B ) should return a string equivalent to A if the relational expression A > B is true.

This suggested implementation applies character comparisons for Max and Min with padding as for relational operators.

For integer or real arrays with no elements (after masking), the Standard currently defines MaxVal as “the negative number of the largest magnitude supported by the processor” and MinVal as “the positive number of the largest magnitude supported by the processor” {and not as Huge, the largest model number}. A similar concept is needed for character arrays with no elements. For the implementation suggested here, the “most negative” value corresponds to a string of characters all equal to Char( 0 ) and the “most positive” value corresponds to a string of characters all equal to Char( n - 1 ) where n is the number of characters in the collating sequence for the kind type parameter of the argument array. Similar terminology appears in the current definition of Char.

Max, Min:

Change “Result Characteristics” to:

Result Characteristics. Type and kind type parameter are the same as the arguments. For arguments of character type, the length of the result is the length of the longest argument.

MaxLoc, MinLoc:

Change “shall be of type integer or real” to “shall be of type integer, real, or character”

MaxVal:

Change “shall be of type integer or real” to “shall be of type integer, real, or character”

Change “The result is of the same type and kind type parameter as Array” to

“The result is of the same type and type parameters as Array”

Replace Result Value Case(i) and Case (ii) as follows (no change to Case(iii)):

Case(i): If the size of Array is not zero, the result of MaxVal( Array ) has a value equal to the maximum value of all the elements of Array. If Array has size zero and type integer or real, the result of MaxVal( Array ) has the value of the negative number of the largest magnitude supported by the processor for numbers of the type and kind type parameter of Array. If Array has size zero and type character, the result of MaxVal( Array ) has the value of a string of characters of length Len( Array ) with each character equal to Char( 0, Kind = Kind( Array ) ).

Case(ii): The result of MaxVal( Array, Mask = Mask ), if Mask has at least one true element, has a value equal to the maximum value of the elements of Array corresponding to true elements of Mask. If Mask has no true elements and the type of Array is integer or real, the result of MaxVal( Array, Mask = Mask ) has the value of the negative number of the largest magnitude supported by the processor for numbers of the type and kind type parameter of Array. If Mask has no true elements and the type of Array is character, the result of MaxVal( Array, Mask = Mask ) has the value of a string of characters of length Len( Array ) with each character equal to Char( 0, Kind = Kind( Array ) ).

MinVal:

Change “shall be of type integer or real” to “shall be of type integer, real, or character”

Change “The result is of the same type and kind type parameter as Array” to

“The result is of the same type and type parameters as Array”

Replace Result Value Case(i) and Case (ii) as follows (no change to Case(iii)):

Case(i): If the size of Array is not zero, the result of MinVal( Array ) has a value equal to the minimum value of all the elements of Array. If Array has size zero and type integer or real, the result of MinVal( Array ) has the value of the positive number of the largest magnitude supported by the processor for numbers of the type and kind type parameter of Array. If Array has size zero and type character, the result of MinVal( Array ) has the value of a string of characters of length Len( Array ) with each character equal to Char(  $n - 1$ , Kind = Kind( Array ) ) where  $n$  is the number of characters in the collating sequence for characters with the kind type parameter of Array.

Case(ii): The result of MinVal( Array, Mask = Mask ), if Mask has at least one true element, has a value equal to the minimum value of the elements of Array corresponding to true elements of Mask. If Mask has no true elements and the type of Array is integer or real, the result of MinVal( Array, Mask = Mask ) has the value of the negative number of the largest magnitude supported by the processor for numbers of the type and kind type parameter of Array. If Mask has no true elements and the type of Array is character, the result of MinVal( Array, Mask = Mask ) has the value of a string of characters of length Len( Array ) with each character equal to Char(  $n - 1$ , Kind = Kind( Array ) ) where  $n$  is the number of characters in the collating sequence for characters with the kind type parameter of Array.

</SUGGESTED IMPLEMENTATION>

<ESTIMATED IMPACT>

Regularizes, simplifies, and increases reliability of many application programs that involve ordering of character data.

</ESTIMATED IMPACT>

<SUBMITTED BY>

Loren Meissner  
2 Kerr Ave  
Kensington CA 94707  
LPMeissner@msn.com

</SUBMITTED BY>

<HISTORY>

<EVENT> February 1996, meeting 136; submitted 96-042

<EVENT> May 1996, meeting 137

<EVENT> August 1996, meeting 138; X3J3 approved

</HISTORY>

</FORTREQ>