

To: J3
 From: Larry Meadows
 Date: August 20, 2000
 Title: Unresolved Issue 265
 Kind parameter of enums

Unresolved issue 265, page 60, raises the problem that the kind parameter of a BIND(C) enum cannot be determined until each enumerator is processed, but it may be necessary to process all of the enumerators before the kind parameter can be determined:

```
ENUM BIND(C) :: MTTYPE
  ENUMERATOR :: I = 1
  ENUMERATOR :: J = I*2
  ENUMERATOR :: K = HUGE(0)
END ENUM MTTYPE
```

The problem is that, in order to evaluate $I*2$, the kind parameter of I must be known, but it isn't known until the kind parameter of K is known.

C guarantees that any enumeration constant shall be representable as an `int`. Therefore, C is able to process the entire enumerated type, evaluating each enumeration constant using `int`, and then determine the integer type used to implement the enumerated type.

A similar algorithm is possible in Fortran. The implementation should select the integer kind that interoperates with the companion C processor's `int` kind, evaluate the enumerators using that integer kind, and then select the integer kind that interoperates with the corresponding C enumerated type.

Note that this requires that the `C_INT` type alias be a valid kind parameter for the Fortran processor. We believe we should require this, since otherwise virtually no interoperability would be possible.

Edits:

[60:4] add "is significant" just before the comma (note: this is unrelated but it is a minor error on the same page, so I put it here).

[60:5-26] Change to a note:

Note 4.xxx

C guarantees that enumeration constants fit in a C `int` (6.7.2.2 of the C standard). Therefore the Fortran processor can evaluate all enumerator values using the integer of kind `C_INT`, and then determine the kind parameter of the integer type that interoperates with the corresponding C enumerated type.

[385:24] Change "The values of `C_INT`," to "The value of `C_INT` shall be a valid value for an integer kind parameter on the processor. The values of"