Proposal New Work Item Language Compatible Mathematical Procedure Standard.

This document in the attachment of the standard ISO cover sheet for the proposal for a new work item.

1. JUSTIFICATION

1.1 Needs

This proposal is a follow-on to the work initiated in the Language Compatible Arithmetic Standard, currently under development in SC22/WG11. This and other follow-on efforts are planned in order to aid members of the scientific community in sharing and exchanging their numerical software. Such scientists use a variety of systems, which differ in many ways, such as:

- In their underlying hardware;
- In the accuracy and reliability of their libraries of math procedures, including conversions between decimal and internal representations;
- In their supporting software, e.g. compiler optimization strategy.

These differences pose difficulties, of which a frequent symptom is a wide variation in the results of a particular computation when executed on different systems. The question then arises as to which, if any, of these widely varying results is reliable. A further question is whether the problem is somewhere in the program, or whether it lies in one or more of the arithmetic environments provided by the systems on which it was executed.

There is a need for a definition of a computing environment which supports the reliable interchange of scientific programs among diverse systems. Standardization is one way to provide this definition.

The first step in providing such an environment is the Language Compatible Arithmetic Standard (work item JTC1.22.28). It defines integer and (real) floating point data types, together with their basic arithmetic operations, including accuracy specifications, and actions to be taken on the occurrence of arithmetic exceptions. It does not cover the properties of sequences of basic operations.

The LCMPS (Language Compatible Mathematical Procedure Standard) is the second step. Its goal is to build upon the arithmetic specifications in the LCAS to provide specifications for the (real) math procedures presently included in the specifications of language and operating system standards. In addition to such functions, it will provide specifications for conversions between decimal and internal representations of arithmetic data.

The extension of these two standards to include complex arithmetic and complex math procedures will be the third step. To some extent, the completion of the first two steps is a prerequisite to an adequate treatment of complex data types because (a) complex arithmetic operations are defined in terms of real arithmetic operations, and (b) the Cartesian and polar representations of a complex number are related by trigonometric functions.

1.1.1 Maintenance of Acceptable Accurary. In the absence of arithmetic exceptions, variations in existing arithmetic implementations usually induce only minor differences in the accuracy produced, and are rarely a major obstacle to the successful interchange of scientific programs. Thus the specifications for the LCAS were expedited by the fact that most of the variations among existing implementations are in exception handling procedures.

In contrast, even in the absence of exceptions, the accuracies of math routine libraries range from quite close to that of the basic arithmetic operations to totally unacceptable. Moreover, the variations in exception handling among the math procedure libraries, provided by various vendors, is believed to be quite large.

The LCMPS will define acceptable accuracy levels, on a procedure-by-procedure basis, where necessary. Because of an inevitable accumulation of rounding errors in the course of a sequence of basic operations, it is likely that the accuracy requirements of this standard will be somewhat less stringent than those implied by the LCAS. In addition, it is necessary that a suitable balance between accuracy and performance be maintained.

The accuracy requirements will appear as error bounds. Of course, an implementor would be free to provide library procedures with smaller bounds. It can therefore be expected that variations in the results produced for a user's program by different systems will still exist.

In the absence of exceptions, wide variations in the results of a user's program in different environments can be expected to occur only for ill-conditioned calculations. In fact, the occurrence of wide variations could be interpreted as a symptom of ill-condition, which can best be remedied by modifying the algorithms used in modeling the original problem to be solved.

1.1.2 Handling of Exceptions. If the input to a procedure is invalid, or if valid input produces an out-of-range result, an error condition exists. The LCMPS will require that all such errors be reported. Sufficient leeway will be permitted in such reporting, so as not to conflict with any language standard.

If no error condition occurs, the procedure will be required to return a valid in-range result.

1.1.3 A Supplement for Language Standards. Most language standards include a list of required intrinsic functions. Some specify the action to be taken for an invalid operand (e.g. log(-1)), but few, if any, contain specifications for actions to be taken on the occurrence of other arithmetic exceptions. It is believed that none provide specifications on the accuracy of the intrinsic functions.

In a sense, therefore, the LCMPS can be regarded as a supplement to language standards.

It may be necessary for the LCMPS to provide means to resolve such conflicts as may exist between language standards.

1.2 Recommended Scope of the Standard

The LCMPS will provide specifications for all intrinsic functions required by those languages in common use for mathematical software, as well as other procedures in widely used run-time libraries.

The LCMPS will also provide specifications for conversions between decimal (text strings) and the internal representations for integer and floating point data types.

The specifications will cover accuracy, exception reporting, and such other matters as are relevant to the reliability of the results returned by mathematical procedures.

The LCMPS will maintain compatibility with all language specifications, and will try to provide means to resolve any conflicts which may exist between language standards.

1.3 Existing Practice in Area of Proposed Standard

It is believed that most systems support one or more standard languages and provide the libraries required by all supported languages. The big problem is in the wide variation in quality among the libraries provided by different vendors.

In addition, all vendors support conversions between decimal and the internal representations of their arithmetic data types.

It is a goal of this standard to limit the variations in the results returned by such software.

1.4 Expected Stability

The specifications in the LCMPS will be independent of any particular arithmetic architecture. Further it is anticipated that its specifications will be largely independent of particular algorithms currently in use. Thus, its useful life should be unaffected by advances in the state of the art.

Because the LCMPS is designed to support portability among diverse systems, it will serve a useful purpose as long as there is significant variation in the arithmetic capabilities offered by different systems. For example, it is important that programmers have the ability to develop software in a workstation environment for execution in a supercomputer environment. It therefore seems likely that the proposed standard would remain stable for ten to twenty years.

2. THE WORK PLAN

A survey will be conducted to determine

1. the mathematical procedures required by language standards and run-time library standards,
2. other mathematical procedures that are in wide use, but are not tied to specific application domains,
3. the constraints imposed on such procedures by formal standards and established usage,
4. the current "state of the art" for implementing these procedures with high accuracy and good performance.

Based on this information, the LCMPS will be produced. It is expected that (at least) the following procedures will be standardized:

1. the 6 trigonometric functions, the corresponding inverse functions, and the two-argument arctangent,
2. the hyperbolic functions,
3. the exponential and logarithmic functions,
4. the "error" function, the "gamma" and "log gamma" functions, and selected Bessel functions,
5. conversion from decimal to all provided floating point types,
6. conversion to decimal from all provided floating point types (including control of the target precision).

The LCMPS will specify the required domains, ranges, and accuracy of these procedures, as well as required error reporting. These specifications will be presented in a style compatible with the LCAS.

3. RELATED STANDARDS EFFORTS

ISO/IEC JCT1/SC22/WG11 is working on the LCAS project. This is done in coordination with X3T2.

ISO/IEC JCT1/SC22/WG9 is currently working on specifications for a standard Ada package of mathematical functions. The accuracy constraints and handling of exceptions are derived from the Ada specifications for arithmetic accuracy and exception handling. Portability of such functions is encouraged, but not required. It appears that the goals of this effort differ from those of the LCPMS.

ISO/IEC JTC1/SC22/WG14 is considering a similar package.