ISO/IEC JTC1/SC22
Languages
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC22

N818

AUGUST 1990

TITLE:	Proposal for a NWI for SC22 on Language Compatible Complex Arithmetic and Procedure Standard and Letter Ballot
SOURCE:	Secretariat JTC1/SC22
WORK ITEM:	N/A
STATUS:	New
CROSS REFERENCE:	N/A
DOCUMENT TYPE:	Proposal for a NWI/Letter Ballot
ACTION:	For action as appropriate by SC22 Member Bodies. SC22 Member Bodies are requested to complete the attached letter ballot and return it to the SC22 Secretariat by 1990-11-23. This document will be discussed at the forthcoming SC22 AG meeting.

Address reply to: ISO IEC JTC 1 SC 22 Secretariat
J L Côté, 140 O'Connor St., 10th Floor
Ottawa, Ont., Canada K1A 0R5
Telephone: (613)957-2496 Telex: 053-3336
Fax: (613) 996-2690

ISO/IEC JTC1/SC22
Languages
Secretariat: CANADA (SCC)

attachment to
JTC1/22 N818

CIRCULATED: 1990-08-13

LETTER BALLOT

From the MEMBER BODY of: _____

On a proposal for a NWI for SC22 on Language Compatible Complex Arithmetic and Procedure Standard.

This Letter Ballot is to be returned by each 'P' Member Body to the SC22 Secretariat of JTC1/SC22 by 1990-11-23.

We support the proposal as presented.

or

We support the proposal, with the attached comments.

or

We do not support the proposal for the technical reasons attached to this ballot.

or

We abstain from voting.
('P' Member Bodies have an obligation to vote)

* DELETE WHICHEVER DOES NOT APPLY.

Place and Date: _____

Signature: _____

Address reply to ISO IEC JTC 1 SC22 Secretariat
J.L. Côté, 140 O'Connor St., 10th Floor
Ottawa, Ont., Canada K1A 0R5
Telephone: (613)957-2496 Telex: 053-3336
Fax: (613) 996-2690

Proposal New Work Item Language Compatible Complex Arithmetic and 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 (LCAS), currently under development in SC22/WG11. The work will be carried out concurrently with the development of the Language Compatible Mathematical Procedure Standard (LCMPS), for which separately a new work item is requested by SC22/WG11. It will extend these two standards to complex data types. These 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:

- 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 is the second step. Its goal is to build upon the arithmetic specifications in the LCAS to provide specifications for the (real) procedures presently included in 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 LCCAPS (Language Compatible Complex Arithmetic and Procedure Standard) will extend the specifications of the LCAS and LCMPS to complex arithmetic and complex math procedures. To some extent, the completion of the LCAS and the LCMPS 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.

All three of the above either are or will be pursued concurrently in the International Organization for Standardization.

1.1.1 Maintenance of Acceptable Accuracy. In the absence of arithmetic exceptions, variations in existing implementations of real arithmetic usually induce only minor differences in the accuracy produced, and are rarely a major obstacle to the successful

interchange of scientific programs.

In contrast, even in the absence of exceptions, the accuracies of math procedure libraries range from quite close to that of the basic arithmetic operations to totally unacceptable. Most systems implement complex arithmetic by procedure calls, and the variation in the results returned (except for complex addition and subtraction) is similar to that seen for other math procedures.

In addition, the variations in exception handling for complex arithmetic (particularly complex division) and the complex math procedures, provided by various vendors, is believed to be quite large.

The LCCAPS will define acceptable accuracy levels. Because of an inevitable accumulation of rounding errors in the course of a sequence of basic operations, it is likely that the accuracy requirements for complex arithmetic and math procedures will be comparable to those of the LCMPS, both being 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.

Most implementations of complex data types are in Cartesian form, consisting of the real and imaginary parts. The polar form of complex data is useful in many applications, for example in electromagnetic theory. In fact, Extended Pascal provides both Cartesian and polar representations for complex. The standard will define acceptable accuracy levels for conversions between the Cartesian and polar representations.

Complex arithmetic operations and also complex math procedures will be based on real arithmetic operations and math procedures, which satisfy the specifications of the LCAS and the LCMPS.

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. Complex arithmetic can be expected to produce the floating point exceptions overflow, underflow and division by zero. There will, however, be many fewer cases for invalid input for complex math procedures than for real. For example, log(-1), arcsin(10), etc. all have well defined results in the complex domain.

If the input to a complex procedure is invalid, or if valid input produces an out-of-range result, an error condition exists. The LCCAPS 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.

The complex functions are often many-valued, and a branch cut will be specified for each such function in order to define a "principal value" so as to maintain consistency among the various functions.

1.1.3 A Supplement for Language Standards. Some language standards include a list of required complex operations and intrinsic functions. Some specify the action to be taken for invalid input. However few, if any, contain specifications for branch cuts. It is

believed that none provide specifications on the accuracy of the complex math procedures.

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

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

1.2 Recommended Scope of the Standard

The LCCAPS will provide specifications for all complex operations and math procedures required by standard languages in common use for mathematical software, as well as other procedures in widely used run-time libraries.

The LCCAPS will also provide specifications for conversions between the Cartesian and polar representations of complex data types.

The specifications will cover accuracy, exception reporting, and such other matters as are relevant to the reliability of the results of complex arithmetic and math procedures.

The LCCAPS 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 complex operations and complex math procedures required by all supported languages. The big problem is in the wide variation in quality among the complex results generated by different vendors.

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 LCCAPS 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 it 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 LCCAPS would remain stable for ten to twenty years.

2. THE WORK PLAN

A survey will be conducted to determine

1. the complex mathematical procedures required by language standards and run-time library standards,
2. other complex 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 LCCAPS will be produced. It is expected that (at least) the following operations and functions will be standardized:

1. the 'normal' complex arithmetic operations and functions, including addition, subtraction, multiplication, division, negation, square-root, real-part, imaginary-part, magnitude ("rho"), argument ("theta"), and the two composition functions: create-from-real-and-imaginary, and create-from-rho-and-theta,
2. the complex trigonometric functions, and the corresponding inverse functions,
3. the complex hyperbolic functions,
4. the complex exponential and logarithmic functions,
5. the complex "gamma" and "log gamma" functions, and selected complex Bessel functions,
6. conversions between complex types, including conversions between Cartesian and polar forms.

The LCCAPS 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 and LCMPS.

The LCCAPS will be developed concurrently with the LCMPS.

3. RELATED STANDARDS EFFORTS

The LCAS is currently under development by ISO/IEC JTC1/SC22/WG11. The LCMPS and LCCAPS are being proposed simultaneously as follow-on projects. The LCCAPS will build on specifications in both the LCAS and LCMPS.

ISO/IEC JCT1/SC22/WG9 is currently working on specifications for a standard Ada package of mathematical functions. Also, a preliminary proposal for complex operations and complex mathematical procedures is under discussion. However, the current focus is on real procedures. The accuracy constraints and handling of exceptions are derived from Ada specifications. Portability is encouraged, but not required. It appears that the goals of this effort will differ from those of the LCCAPS.

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