



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

ISO/IEC JTC1/SC22
N 1010

AUGUST 1991

TITLE: Summary of Voting and Comments Received
on CD 10967: Information Technology -
Programming Languages - Language Compatible
Arithmetic

SOURCE: Secretariat ISO/IEC JTC1/SC22

WORK ITEM: JTC1.22.28

STATUS: New

CROSS REFERENCE: N935

DOCUMENT TYPE: Summary of Voting - CD

ACTION: For information to SC22 Member Bodies.
See attached.

SUMMARY OF VOTING ON:

Letter Ballot Reference No: SC22 N935
Circulated by :JTC1/SC22
Circulation Date :1991-03-28
Closing Date :1991-07-18

SUBJECT: CD 10967: Information Technology - Programming Languages -
Language Compatible Arithmetic

The following responses have been received:

'P' Members supporting the proposal,
without comments : 05

'P' Members supporting the proposal,
with comments : 03

'P' Members not supporting the proposal : 02

'P' Members abstaining : 00

'P' Members not voting : 09

Comments:

Attachment 1 - Czechoslovakia
Attachment 2 - France
Attachment 3 - Japan
Attachment 4 - UK
Attachment 5 - USA

Secretariat Action:

The SC22 Secretariat will forward the comments to WG11 for recommendation on further processing of CD 10967.

ISO/IEC JTC1/SC22 LETTER BALLOT SUMMARY

PROJECT NO: JTC1.22.28

SUBJECT: CD 10967: Information Technology - Programming languages - Language compatible arithmetic

Reference Document No: N935
Circulation Date: 1991-03-28

Ballot Document No: N935
Closing Date: 1991-07-18

Circulated To: SC22 P,O,L

Circulated By: Secretariat

SUMMARY OF VOTING AND COMMENTS RECEIVED

	Approve	Disapprove	Abstain	Comments	Not Voting
<u>'P' Members</u>					
Belgium	()	()	()	()	(/)
Brazil	()	()	()	()	(/)
Canada	(x)	()	()	()	()
China	()	()	()	()	(/)
Czechoslovakia	(x)	()	()	(x)	()
Denmark	()	()	()	()	(/)
Finland	()	()	()	()	(/)
France	()	(x)	()	(x)	()
Germany	(x)	()	()	()	()
Iran	()	()	()	()	(/)
Italy	()	()	()	()	(/)
Japan	(x)	()	()	(x)	()
Netherlands	(x)	()	()	()	()
New Zealand	()	()	()	()	(/)
Sweden	(x)	()	()	()	()
Switzerland	()	()	()	()	(/)
UK	(x)	()	()	(x)	()
USA	()	(x)	()	(x)	()
USSR	(x)	()	()	()	()
<u>'O' Members</u>					
Australia	()	()	()	()	()
Austria	()	()	()	()	()
Cuba	()	()	()	()	()
Hungary	()	()	()	()	()
Iceland	()	()	()	()	()
India	()	()	()	()	()
Korea	()	()	()	()	()
Poland	()	()	()	()	()
Portugal	()	()	()	()	()
Singapore	()	()	()	()	()
Turkey	()	()	()	()	()
Thailand	()	()	()	()	()
Yugoslavia	()	()	()	()	()

attachment 1
To N1010

Czechoslovak comments on Document
ISO/IEC JTC 1 CD 10 967

The Czechoslovak National Committee is in favour with ISO/JTC 1 CD 10967, however to refer to the Czechoslovak comments sent to N 796.

**AFNOR VOTE ON ISO/IEC JTC1/SC22/N935 (CD 10967)
LANGUAGE COMPATIBLE ARITHMETIC**

The AFNOR vote would be turned into "yes" if the following comment were accepted.

Page 2, 2, Conformity, change first paragraph to read:

A Standard Language implementation conforms to this International Standard:

- if the Language Standard supports at least one signed integer type and/or at least one floating point type, and
- if the implementation of those types are provided in a way that satisfies all the requirements of clauses 4 through 7, for those operations defined in the Language Standard.

W
Z N1010

Japanese Comments on CD 10967(SC 22N935)

Binding for COBOL should be included in Annex B (Bindings for Specific Languages) of CD 10967.

Bindings for many standardized languages are described in Annex B, but the one for COBOL is not.

JUL 30 91 8:24 ALLIED-SIGNAL KGD 516-997-3331

P. 6/ 5

*Attachment 5
To W1010*

The U.S. votes NO on JTC1/SC22 N935, with the following comments. These comments must be satisfactorily resolved for the U.S. to change the U.S. vote to YES.

It was the consensus of WG11 in the meeting of January 1991, in resolving comments by France on the relationship between the LCAS and two New Work Items, that this document would be recast as the first part of a multi-part standard, pending SC22 approval. The U.S., therefore, believes that this document should not be advanced to DIS in its current form, or at least not until SC22 has considered the multi-part standard proposal and the appropriate modifications, if any, to the Scope of this document have been made.

Further, public comments have raised a number of concerns in the areas of notification, rounding, and compatibility with ISO/IEC 559 (IEEE 754). In light of this, the U.S. believes that significant work remains to be done before progressing the LCAS to DIS.



*Attachment 4
to W1010*

The UK endorses the attached comments for consideration by
SC 22/WG 11.

JMS/jms
19 June 1991

SENT 06 JUN 1991

BSI	PRIVATE CIRCULATION
	Doc. No. 91/64517
	Date 2-6-91
	C'tee ref. IST/S
	C'tee title IST/S: 252
Action/Information	

Comments on LCAS 3.1

Roger Scowen
National Physical Laboratory,
Teddington, Middlesex, England.
E-Mail: rss@seg.npl.co.uk

May 10, 1991

1 Introduction

These comments on LCAS – Version 3.1 (= SC22 N935 = SC22/WG11 N229) arise from a study while adopting it in draft standard Prolog (SC22 WG17 N72 - in press). Note that I have not studied the annexes in detail.

I would support LCAS – Version 3.1 as a DIS and have found few problems or errors.

2 Error

At the end of clause 4.3 (Conversion operations) the text "... where $rnd_{F \rightarrow I}$ is a rounding function from F to I ..." should read "where $rnd_{F \rightarrow I}$ is a rounding function from \mathcal{R} to \mathcal{Z} ..." because the range of a rounding function is defined to be \mathcal{R} .

Later (5.9) I suggest giving this operation a more precise definition.

3 Indicate changes

Please indicate clearly in each new draft, the changes which have been made since the last draft. Note that the summary in "Project milestones" on page 20 is too easy to overlook and too brief. As you produce new drafts, I need to change the draft Prolog standard, and I do not wish to compare the previous and current versions of LCAS line by line in order to find the alterations.

I found the following differences in version 3.1:

- (1) In clause 3.2 (Definitions) the definition of "signature" has been added.
- (2) In clause 4.2.4, the addition of a $rnd_and_chk_F(x)$ function, and consequent changes.

I am concerned that I may have missed more subtle changes.

4 Misprints

I have found the following misprints and minor faults:

1. Paragraph 3 of the scope is unnecessary. It repeats a definition in 3.2. Delete it, or replace it by a note cross-referencing the definition.
2. In clause 3.2 (Definitions), comments such as “See F_D in 4.2.” should be notes. Notes and examples should also be notes.
3. In 1.1 Note, Replace “Please see A.1.3 for” by “A.1.3 describes”.
4. In 3.2 arithmetic data type: delete “ (the complex numbers)”. The meaning of \mathcal{Z} , \mathcal{R} and \mathcal{C} are all defined in 3.1.
5. The first note of clause 7 is unnecessary. It repeats a definition in 3.2. Delete it, and perhaps add a cross-reference after “implementation” in the first line.

5 Suggestions

5.1 Define “Range checking function” explicitly

The standard would benefit from the additional definition in clause 3.2:

Range checking function: A function whose signature is:

$$chk_F : \mathcal{R} \times F^{\circ} \rightarrow F \cup \{\text{overflow}, \text{underflow}\}$$

A range checking function $chk_F(x, y)$ determines the final result of a floating point operation based on a before-rounding value x and an after-rounding value y .

NOTE — The permissible range checking functions are defined in clause 4.2.4.

5.2 Signature

It seems better to move the example of a signature from the start of clause 4 to 3.2 Definitions. And in 4 replace the final paragraph and example by: “Each operation is defined by a signature and one or more axioms.”

I therefore prefer the following definition of “signature”:

Signature: A specification of an operation or function which defines its name, and the type of its argument(s) and result.

NOTE — The operation or function is further defined by one or more axioms.

For example, the signature:

$$add_I : I \times I \rightarrow I \cup \{\text{overflow}\}$$

defines the operation add_I which takes two integer arguments ($I \times I$) and produces either a single integer result (I) or the exceptional value overflow.

5.3 Operations and functions

The difference between operations and functions is not clear in LCAS; it should be made explicit. LCAS uses signature to define “operations”, not “functions” except in 4.2.3, 4.2.4, and 4.2.5. But I cannot see why these are functions rather than operations. Note that at the start of clause 4, an arithmetic type is defined as a set of values and operations (but not functions).

5.4 Shall and should

The definitions for “shall” and “should” are unnecessary in an ISO/IEC standard. (See “Rules for the drafting and presentation of International Standards” ISO, 1986, ISBN 92-67-01042-5). Your rules are also phrased differently from the ISO rules: this must be a “bad thing”, even (especially) if they are almost the same.

5.5 signi

On the rare occasions I have required the sign of a value, I have wanted an integer result. I therefore suggest an additional operation:

$$\text{sign}_I: I \rightarrow I$$

where

$$\begin{aligned}\text{sign}_I(n) &= 1 \text{ if } n \geq 0 \\ &= -1 \text{ if } n < 0\end{aligned}$$

5.6 Integer mod

I do not like the fudging in giving alternative definitions. I prefer giving one (either – I am not fussy), or both (with different names).

5.7 Round and check function

The new definition of this function is imprecisely defined by the text: “This combination ... is captured by”. I suggest it is replaced by: “It is therefore simpler to use the function defined by the signature and axiom:”.

5.8 Conversion operation I to F

Conversion operations are very vague compared with the other operations defined by the standard. It cannot be desirable that converting an integer value

$$= -\lfloor |x| \rfloor \quad \text{if } x < 0$$

$$\text{round}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) = \lfloor x + 1/2 \rfloor$$

$$\text{ceiling}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) = -\lfloor -x \rfloor$$

The conversion operations are

$$\text{floor}_{F \rightarrow I} : F \rightarrow I \cup \{\text{overflow}\}$$

$$\text{truncate}_{F \rightarrow I} : F \rightarrow I \cup \{\text{overflow}\}$$

$$\text{round}_{F \rightarrow I} : F \rightarrow I \cup \{\text{overflow}\}$$

$$\text{ceiling}_{F \rightarrow I} : F \rightarrow I \cup \{\text{overflow}\}$$

For all values x in F , and n in I , the following axioms shall apply:

$$\begin{aligned} \text{floor}_{F \rightarrow I}(x) &= \text{floor}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) && \text{if } \text{floor}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \in I \\ &= \text{overflow} && \text{if } \text{floor}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \notin I \end{aligned}$$

$$\begin{aligned} \text{truncate}_{F \rightarrow I}(x) &= \text{truncate}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) && \text{if } \text{truncate}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \in I \\ &= \text{overflow} && \text{if } \text{truncate}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \notin I \end{aligned}$$

$$\begin{aligned} \text{round}_{F \rightarrow I}(x) &= \text{round}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) && \text{if } \text{round}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \in I \\ &= \text{overflow} && \text{if } \text{round}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \notin I \end{aligned}$$

$$\begin{aligned} \text{ceiling}_{F \rightarrow I}(x) &= \text{ceiling}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) && \text{if } \text{ceiling}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \in I \\ &= \text{overflow} && \text{if } \text{ceiling}_{\mathcal{R} \rightarrow \mathcal{Z}}(x) \notin I \end{aligned}$$

An implementation shall provide all four, as in the case of Fortran '90.

6 Annex B.10 Prolog

This annex should be changed to correct the following errors and changes since December 1990. Further changes may well occur as WG17 considers and revises its drafts.

1. Reference 20: change "N64" to "N72".
2. Replace "WG17 has resolved ... floating point type, float." by "Standard Prolog will define (and standard conforming implementations must provide) one integer type, integer, and one floating point type, float, which both conform to LCAS."
3. Replace "When *bounded* is false" by "The value of *minint* shall not be zero, and when *bounded* is false".

4. Replace (twice) "the following identifiers" by "the following flags".
5. Replace "rem(X,Y)" by "x rem y".
6. Replace "A rounding flag is proposed to convey" by "A rounding flag conveys".
7. Replace "float_sign(X)" by "sign(X)".
8. Replace "float_fraction(X)" by "float_significand(X)".
9. Replace "The rounding in $cv_{F \rightarrow I}$ to be defined" by "The rounding in $cv_{F \rightarrow I}$ will be defined as in LISP."
10. Replace "fix(X)" by "floor(X), truncate(X), round(X), ceiling(X)".