INTERNATIONAL STANDARD

ISO/IEC 10514-1

First edition
1996-06-01

Information technology — Programming languages —

Part 1:
Modula-2, Base Language

Technologies de l'information — Langages de programmation —

Partie 1: Modula-2, langage de base

Reference number
ISO/IEC 10514-1:1996(E)

Contents

Annexes

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 10514-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces.*

ISO/IEC 10514 consists of the following parts, under the general title *Information technology — Programming languages*:

— *Part 1: Modula-2, Base Language*

— *Part 2: Object-oriented Modula-2*

Annexes A and B form an integral part of this part of ISO/IEC 10514. Annexes C to F are for information only.

Introduction

This part of ISO/IEC 10514, Modula-2 (Base language), provides a specification of the form and meaning of Modula-2 programs, and by reference to that specification lays down requirements for Modula-2 implementations.

Modula-2 is a general-purpose programming language, characterized by modules that provide data and procedure encapsulation and hiding. It provides facilities for concurrent programming based on coroutines, and includes facilities for low-level programming. Modula-2 is, therefore, suitable for systems programming. Facilities such as input/output, mathematical operations and string processing are made available through library modules.

Modula-2 is a descendent of Modula whose ancestor is Pascal. All three languages were devised by Professor Niklaus Wirth, ETH, Zürich, Switzerland. The first description of Modula-2 was published in 1980. The language became popular during the early 1980s and many implementations were developed. Because of different interpretations of various descriptions of Modula-2, significant differences in implementations appeared, and in 1984 a group of Modula-2 proponents formed a BSI panel to develop a standard. The BSI panel successfully proposed a new work item to ISO, and the ISO working group met for the first time in 1987. In 1989 the first draft of this standard was published, and in 1992 the second committee draft was published. The base documents for the formulation of this part of ISO/IEC 10514 were the various editions of *Programming in Modula-2* by Niklaus Wirth, published by Springer-Verlag.

This part of ISO/IEC 10514 contains a formal model of Modula-2; for the sake of generality, consistency, and precision, differences have been introduced between the model and the third edition of Professor Wirth's book *Programming in Modula-2* and these are identified either as 'clarifications' or 'changes'. During the standardization of Modula-2 it was recognized that users of the language were demanding extensions to the language devised by Professor Wirth and that implementors were responding by providing them. Several changes and extensions have, therefore, been made in order to prevent further divergence, and to ensure consistency and coherence.

This part of ISO/IEC 10514 uses a variety of notations to specify the requirements contained in it. A standard syntactic metalanguage is used for specifiying lexical and syntactic structures. The Vienna Development Method-Specification Language (VDM-SL), which is based upon conventional mathematics, is used to define the semantics of the language in a mathematically rigorous manner. In addition, the semantics of the Modula-2 language, system modules and standard library are also specified using natural language, albeit in a form that is possibly ambiguous. It is intended that the corresponding VDM-SL resolves any such ambiguity. If the natural language requirements and those expressed in the VDM-SL conflict, then there is a mistake in this standard. In many cases, small details of the language semantics may only be resolved by reference to the VDM-SL; by this means the natural language description has been kept relatively simple. A number of typographic and stylistic conventions have been adopted in order to make the natural language description succinct.

The definitions of key terms used in this part of ISO/IEC 10514 are expressed in natural language, with references to VDM-SL where appropriate. The requirements for conforming implementations are given in natural language, with references to VDM-SL where appropriate.

The development of large parts of the formal specification of the semantics of Modula-2 in this part of ISO/IEC 10514 was sponsored by the U.K. Alvey Research Programme. The Alvey sponsored work was carried out at Leicester University Computer Studies Department in collaboration with BSI QA Software Engineering Department.

Information technology — Programming languages — Part 1: Modula-2, Base Language

Section 1 : Scope

1.1 Goals

The goals of this part of ISO/IEC 10514 are:

— to provide a rigorous definition of the language Modula-2 and its standard library by providing a mathematical model of both;

— to provide a resolution of differences among interpretations of other descriptions of Modula-2 and its standard library, while endeavouring to preserve investment in existing practice;

— to remove features thought to be redundant, inherently flawed, or inadequate;

— to specify new language and standard library facilities where a need is perceived to exist;

— to maintain the general principles of Modula-2 laid down by its inventor, while allowing for later modernization and standardization.

1.2 Specifications included in this part of ISO/IEC 10514

This part of ISO/IEC 10514 provides specifications for:

— required symbols for Modula-2 program representation, including comments, literals, and source code directives;

— the lexical structure, the syntactic structure and the semantics of Modula-2 programs, including programs that make use of system modules.

— the interface to and the semantics of standard Modula-2 library modules.

— those separate modules of the standard library that a conforming implementation is required to supply;

— violations of the rules for use of the language, system modules and standard library modules that a conforming implementation is required to detect;

— certain criteria for the size and complexity of programs that a conforming implementation must accept;

— further compliance requirements for implementations, including documentation requirements.

1.3 Specifications not within the scope of this part of ISO/IEC 10514

This part of ISO/IEC 10514 provides no specifications for:

— the underlying representation of predefined data types (except in the case of packedset types; see 7.1.7.1);

— the method by which implementations are invoked (including identification of the program module and associated definition and implementation modules);

— the method by which compilation modules are stored (including the correspondence between module names and system file names where files are used);

— the method by which implementations accept input (including the encoding of source text and including the number of compilation modules accepted for each invocation);

— performance aspects of implementations, and certain quality aspects not covered by 1.2;

— the effect of executing a program that uses extensions to the language, extensions to system modules or extensions to standard library modules, or that otherwise deviates from this part of ISO/IEC 10514;

— the effect of continuing execution of a program in which an exception has occurred and execution has continued without an exception being raised;

— the meaning of a program that relies on a definition of implementation-dependent values or implementation-dependent behaviour.

Section 2 : Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 10514. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 10514 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 10967-1:1994, *Information technology—Language independent arithmetic—Part 1: Integer and floating point arithmetic.*

ISO/IEC 13817-1:...[1)], *Information technology — Programming languages, their environments and system software interfaces — Vienna Development Method — Specification Language—Part 1: Base language.*

BS 6154:1981, *Method of defining Syntactic Metalanguage.*

IEC 559:1989, *Binary floating-point arithmetic for microprocessor systems*

[1)] To be published.