TECHNICAL REPORT

# ISO/IEC TR 24731-2

First edition
2010-12-01

# Information technology — Programming languages, their environments and system software interfaces — Extensions to the C library —

## Part 2: Dynamic Allocation Functions

*Technologies de l'information — Langages de programmation, leurs environnements et leurs systèmes d'interface de logiciel — Extensions à la bibliothèque C —*

*Partie 2: Fonctions d'attribution dynamiques*

Reference number
ISO/IEC TR 24731-2:2010(E)



ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

# Contents



 

# 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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. 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.

In exceptional circumstances a technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when a technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 24731-2, which is a Technical Report of type 2, 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 TR 24731 consists of the following parts, under the general title *Information technology—Programming languages, their environments and system software interfaces—Extensions to the C library*:

— Part 1: Bounds-checking interfaces

— Part 2: Dynamic Allocation Functions

 

# Introduction

Traditionally, the C library has contained many functions that trust the programmer to provide output character arrays big enough to hold the result being produced. Not only do these functions not check that the arrays are big enough, they frequently lack the information needed to perform such checks. While it is possible to write safe, robust, and error-free code using the existing library, the library tends to promote programming styles that lead to mysterious failures if a result is too big for the provided array.

Perhaps the most common programming style is to declare character arrays large enough to handle most practical cases. However, if the program encounters strings too large for it to process, data is written past the end of arrays overwriting other variables in the program. The program never gets any indication that a problem exists, and so never has a chance to recover or to fail gracefully.

Worse, this style of programming has compromised the security of computers and networks. Daemons are given carefully prepared data that overflows buffers and tricks the daemons into granting access that should be denied.

If the programmer writes run time checks to verify lengths before calling library functions, then those run time checks frequently duplicate work done inside the library functions, which discover string lengths as a side effect of doing their job.

ISO/IEC TR 24731 provides alternative functions for the C library that promote safer, more secure programming. ISO/IEC TR 24731-1 provides simple replacement functions for the library functions of ISO/IEC 9899:1999 that provide bounds checking. Those function can be used as simple replacements for the original library functions in legacy code. This part of ISO/IEC TR 24731 presents replacements for many of these functions that use dynamically allocated memory to ensure that buffer overflow does not occur. Since the use of such functions requires adding additional calls to free the buffers later, these functions are better suited to new developments than to retrofitting old code.

In general, the functions described in this part of ISO/IEC TR 24731 provide much greater assurance that buffer overflow problems will not occur, since buffers are always automatically sized to hold the data required. With the bounds checking functions, if an invalid size was passed to one of the functions, it could still suffer from buffer overflow problems, while appearing to have addressed such issues. Applications that use dynamic memory allocation might, however, suffer from denial of service attacks where data is presented until memory is exhausted.

These functions are drawn from existing implementations that have widespread usage. Many of these functions are included in ISO/IEC 9945:2003 (POSIX) and as such are aligned with that International Standard.

 

Many of the interfaces in this part of ISO/IEC TR 24731 are derived from interfaces specified in other ISO/IEC International Standards, and in particular ISO/IEC 9945:2003 (including Technical Corrigendum 1), and ISO/IEC 23360:2006.

Where an interface is described as being derived from either of these International Standards, the functionality described on this reference page is intended to be aligned with that International Standard. Any conflict between the requirements described in this part of ISO/IEC TR 24731 and the referenced International Standard is unintentional. This part of ISO/IEC TR 24731 defers to the underlying International Standard.

 

# Information technology — Programming languages, their environments and system software interfaces — Extensions to the C library —

## Part 2:
## Dynamic Allocation Functions

## 1. Scope

ISO/IEC TR 24731 specifies a series of extensions of the programming language C, specified by ISO/IEC 9899:1999. ISO/IEC 9899:1999 provides important context and specification for this part of ISO/IEC TR 24731. Clause 4 should be read as if it were merged into ISO/IEC 9899:1999, 6.10.8. Clause 5 should be read as if it were merged into the parallel structure of named subclauses of ISO/IEC 9899:1999, Clause 7.

## 2. Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 9899:1999, *Programming languages—C*.

ISO/IEC 9899:1999/Cor-1:2001, *Programming languages—C—Technical Corrigendum 1*.

ISO/IEC 9899:1999/Cor-2:2004, *Programming languages—C—Technical Corrigendum 2*.

ISO/IEC 9899:1999/Cor-3:2007, *Programming languages—C—Technical Corrigendum 3*.

ISO/IEC 23360:2006, *Linux standard Base (LSB) core specification 3.1*

 