

Errata for SEI CERT C Coding Standard (2016 Edition)

This list of errata lists errors identified since the release of the PDF document entitled [SEI CERT C Coding Standard: Rules for Developing Safe, Reliable, and Secure Systems \(2016 Edition\)](#). These corrections were made to the [SEI CERT C Coding Standard wiki](#), which is the most current source for this standards information. The focus of the correction is highlighted in **red** below in the Before and After columns. This page does not include additions since this standard was published, if the addition doesn't actually correct a defect that was in the publication.

Location	Before (with error)	After (with correction)	Rationale
p. 30, Section 2.3.1);	}	The last line of the Noncompliant Code Example in section 2.3.1 closes the code block incorrectly with a close-parent-semicolon ");", but should be a closed brace "}".
p. 150, Section 5.3.5.2	This compliant solution eliminates signed overflow on systems where long is at least twice the precision of int:	This compliant solution eliminates signed overflow on systems where long long is at least twice the precision of int:	The phrase should be "long long" not "long."
p. 189, Section 6.4.1	<code>long int big = 1234567890;</code>	<code>long int big = 1234567890L;</code>	Addition of type suffix in initialization statement.
p. 190, Section 6.4.2	<code>long int big = 1234567890;</code>	<code>long int big = 1234567890L;</code>	Addition of type suffix in initialization statement.
p. 203, Section 7.2.1	In this noncompliant code example, a variable length array of size is declared.	In this noncompliant code example, a variable length array of size size is declared.	The second "size" was missing.
p. 226, Section 8.1	A <i>character string literal</i> is a sequence of zero or more multibyte characters enclosed in double-quotes, as in " xyz ."	A <i>character string literal</i> is a sequence of zero or more multibyte characters enclosed in double-quotes, as in " xyz ".	The period should appear outside the quote, not inside the quote because the quote is part of the character string.
p. 273 Section 9.5.1	The code checks for unsigned integer overflow in compliance with INT32-C. Ensure that operations on signed integers do not result in overflow and also ensures that len is not equal to zero.	The code attempts to check for unsigned integer overflow in compliance with INT30-C. Ensure that unsigned integer operations do not wrap and also ensures that len is not equal to zero.	The code uses only unsigned integers.
p. 286, Section 10.2.2	When opening a FIFO with O_RDONLY or O_WRONLY set: When opening a block special or character special file that supports nonblocking opens: Otherwise, the behavior of O_NONBLOCK is unspecified.	When opening a FIFO with O_RDONLY or O_WRONLY set: <ul style="list-style-type: none">• If O_NONBLOCK is set, an <code>open()</code> for reading-only returns without delay. An <code>open()</code> for writing-only returns an error if no process currently has the file open for reading.• If O_NONBLOCK is clear, an <code>open()</code> for reading-only blocks the calling thread until a thread opens the file for writing. An <code>open()</code> for writing-only blocks the calling thread until a thread opens the file for reading. When opening a block special or character special file that supports nonblocking opens: <ul style="list-style-type: none">• If O_NONBLOCK is set, the <code>open()</code> function returns without blocking for the device to be ready or available; subsequent behavior is device-specific.• If O_NONBLOCK is clear, the <code>open()</code> function blocks the calling thread until the device is ready or available before returning. Otherwise, the behavior of O_NONBLOCK is unspecified.	Bulleted items were missing.
p. 365, Section 12.2.3	Signal handlers can refer to objects with static or thread storage a duration that are lock-free atomic objects, as in this compliant solution:	Signal handlers can refer to objects with static or thread storage durations that are lock-free atomic objects, as in this compliant solution:	The phrase should be "storage durations" not "storage a duration."
p. 447, Section 14.11.1	This noncompliant code example declares a shared <code>atomic_bool flag</code> variable and provides a <code>toggle_flag()</code> method that negates the current value of <code>flag</code> :	This noncompliant code example declares a shared <code>atomic_bool flag</code> variable and provides a <code>toggle_flag()</code> method that negates the current value of <code>flag</code> :	The variable should be "atomic_bool flag," not "atomic_boolflag."

p. 447, Section 14.11.1	Execution of this code may result in a <u>data race</u> because the value of <code>flag</code> is read, negated, and written back.	Execution of this code may result in unexpected behavior because the value of <code>flag</code> is read, negated, and written back.	Changed "data race" to a more appropriate term.
-------------------------------	--	--	---