

The charts below are based on `au-misra3.lnt`, our configuration file for MISRA 2012 which is for use with PC-lint / FlexeLint version 9.00k and later. Although we have taken great care in creating this chart, we make no guarantees as to its completeness and accuracy. If you find any discrepancies or errors, please send an email to "sales" at "gimpel.com" with the details.

MISRA C 2012 Directives

	Fully Supported	Partially Supported	Not Statically Checkable	Not Yet Checked	Totals
Required Rules	2	3	4	0	9
Advisory Rules	4	2	1	0	7
Totals	6	5	5	0	16

Directive	Category	Decidable	Supported	Notes
1.1	req	No	n/a	Not Statically Checkable.
2.1	req	No	n/a	Not Statically Checkable.
3.1	req	No	n/a	Not Statically Checkable.
4.1	req	No	n/a	Not Statically Checkable.
4.2	adv	No	n/a	Not Statically Checkable.
4.3	req	No	Partial	Rule is not decidable but assistance is provided.
4.4	adv	No	Yes	Rule deemed not decidable but checking is supported.
4.5	adv	No	Yes	Rule deemed not decidable but checking is supported.
4.6	adv	No	Yes	Rule deemed not decidable but checking is supported.
4.7	req	No	Yes	Rule deemed not decidable but checking is supported.
4.8	adv	No	Yes	Rule deemed not decidable but checking is supported.
4.9	adv	No	Partial	Rule is not decidable but assistance is provided.
4.10	req	No	Yes	Rule deemed not decidable but checking is supported.
4.11	req	No	Partial	Rule is not decidable but assistance is provided.
4.12	req	No	Partial	Rule is not decidable but assistance is provided.
4.13	adv	No	Partial	Rule is not decidable but assistance is provided.

MISRA C 2012 Rules

	Decidable			Undecidable			Totals
	Fully Supported	Partially Supported	Not Yet Checked	Fully Supported	Partially Supported	Not Yet Checked	
Required Rules	44	31	8	6	8	4	101
Advisory Rules	17	8	3	2	1	1	32
Mandatory Rules	3	1	0	2	2	2	10
Totals	65	39	10	10	11	7	143

Rule	Category	Decidable	Supported	Notes
1.1	req	No	Partial	Rule is not decidable but assistance is provided.
1.2	adv	No	Partial	Rule is not decidable but assistance is provided.
1.3	req	No	n/a	Not Statically Checkable.
2.1	req	No	Yes	Rule deemed not decidable but checking is supported.
2.2	req	No	Yes	Rule deemed not decidable but checking is supported.
2.3	adv		Yes	
2.4	adv		Partial	Instances where tag name appears only inside a <code>typedef</code> are not reported.
2.5	adv		Yes	
2.6	adv		Yes	
2.7	adv		Yes	
3.1	req		Partial	C-style comment inside of c++-style comment not reported.
3.2	req		Yes	
4.1	req		Yes	
4.2	adv		Yes	
5.1	req		Yes	
5.2	req		Yes	
5.3	req		Yes	
5.4	req		Yes	
5.5	req		No	
5.6	req		Partial	Uniqueness is not checked across all scopes and name spaces.
5.7	req		Partial	Uniqueness is not checked across all scopes and name spaces.
5.8	req		No	
5.9	adv		Partial	Uniqueness is not checked across all name spaces and translation units.
6.1	req		Yes	
6.2	req		Yes	
7.1	req		Yes	
7.2	req		Yes	
7.3	req		Yes	
7.4	req		Partial	Direct modification of a string literal is not reported.
8.1	req		Yes	
8.2	req		Yes	

Rule	Category	Decidable	Supported	Notes
8.3	req		Partial	Differing names and top-level qualifiers in function declarations are not reported.
8.4	req		No	
8.5	req		Yes	
8.6	req		Yes	
8.7	adv		Yes	
8.8	req		Yes	
8.9	adv		Yes	
8.10	req		Yes	
8.11	adv		No	
8.12	req		Partial	False positive emitted for <code>enum</code> value explicitly repeated.
8.13	adv	No	Yes	Rule deemed not decidable but checking is supported.
8.14	req		Yes	
9.1	mand	No	Partial	Rule is not decidable but assistance is provided.
9.2	req		Partial	The designated initializer and <code>= {0}</code> exceptions are not supported resulting in false positives for these constructs.
9.3	req		Partial	False positive due to lack of support for sparse initialization exception. Additionally, a message is issued for all aggregate types, not just arrays.
9.4	req		Yes	
9.5	req		Yes	
10.1	req		Partial	No diagnostic is produced for using a non-boolean type as the condition of a ternary operator. The exception allowing non-negative ICE on the RHS of a shift expression is not supported.
10.2	req		Yes	
10.3	req		Partial	A false positive occurs when the value of the macros <code>true</code> or <code>false</code> are assigned to a boolean type.
10.4	req		Yes	
10.5	adv		Partial	A false positive occurs when the value of the macros <code>true</code> or <code>false</code> are cast to a boolean type.
10.6	req		Yes	
10.7	req		Yes	
10.8	req		Yes	
11.1	req		Partial	Conversions between two function pointers of differing type are not reported.
11.2	req		No	
11.3	req		Partial	Adding type qualifiers in a cast results in a false positive. The exception allowing casting to a char pointer is not supported.
11.4	adv		Partial	Various false positives and false negatives.
11.5	adv		No	
11.6	req		Yes	
11.7	req		Yes	
11.8	req		Yes	
11.9	req		Partial	Comparing a pointer to 0 via <code>==</code> , <code>!=</code> , or the ternary operators is not reported.
12.1	adv		Partial	False positives and false negatives from edge cases.

Rule	Category	Decidable	Supported	Notes
12.2	req	No	Yes	Rule deemed not decidable but checking is supported.
12.3	adv		Yes	
12.4	adv		Partial	False positive when overflow occurs during the evaluation of <code>const</code> expression.
13.1	req	No	Yes	Rule deemed not decidable but checking is supported.
13.2	req	No	Partial	Rule is not decidable but assistance is provided.
13.3	adv		Partial	An expression of the form <code>*p++</code> yields a false positive.
13.4	adv		Partial	Assignment occurring inside array indexing and function calls not reported.
13.5	req	No	Yes	Rule deemed not decidable but checking is supported.
13.6	mand		Partial	The use of <code>sizeof</code> with a VLA type is not reported.
14.1	req	No	Partial	Rule is not decidable but assistance is provided.
14.2	req	No	No	
14.3	req	No	Partial	Rule is not decidable but assistance is provided.
14.4	req		Partial	A false positive is produced when the controlling expression consists of the macro <code>true</code> or <code>false</code> .
15.1	adv		Yes	
15.2	req		No	
15.3	req		Partial	No diagnostic is produced when a label defined in a <code>switch</code> clause that is not a compound statement is referenced by a <code>goto</code> in a different <code>switch</code> clause for the same <code>switch</code> .
15.4	adv		No	
15.5	adv		Yes	
15.6	req		Partial	Unbraced bodies of <code>if</code> statements are not diagnosed.
15.7	req		Partial	If an <code>else</code> statement is present, no message is produced regardless of whether the <code>else</code> contains a comment or side effect.
16.1	req		No	
16.2	req		Yes	
16.3	req		Partial	No diagnostic is produced when the last switch-clause does not contain a <code>break</code> statement.
16.4	req		Partial	No diagnostic is produced when the <code>switch</code> type is an enumeration. No diagnostic is produced if the <code>default</code> label does not contain a statement or a comment.
16.5	req		Partial	Only the presence of a <code>default</code> label is checked, not its position.
16.6	req		Partial	Only <code>switch</code> statements without <code>case</code> labels are reported.
16.7	req		Yes	
17.1	req		Partial	The use of the <code>va_list</code> type is not diagnosed.
17.2	req	No	No	
17.3	mand		Yes	
17.4	mand		Yes	
17.5	adv	No	No	
17.6	mand		Yes	
17.7	req		Yes	
17.8	adv	No	Yes	Rule deemed not decidable but checking is supported..
18.1	req	No	Partial	Rule is not decidable but assistance is provided.
18.2	req	No	Partial	Rule is not decidable but assistance is provided.

Rule	Category	Decidable	Supported	Notes
18.3	req	No	Partial	Rule is not decidable but assistance is provided.
18.4	adv		Yes	
18.5	adv		Yes	
18.6	req	No	Partial	Rule is not decidable but assistance is provided.
18.7	req		Yes	
18.8	req		Yes	
19.1	mand	No	No	
19.2	adv		Yes	
20.1	adv		Yes	
20.2	req		Yes	
20.3	req		Partial	Multiple string tokens after an <code>#include</code> are not reported.
20.4	req		Yes	
20.5	adv		Yes	
20.6	req		Yes	
20.7	req		Yes	
20.8	req		Yes	
20.9	req		Yes	
20.10	adv		Yes	
20.11	req		Yes	
20.12	req		Yes	
20.13	req		Yes	
20.14	req		Partial	Only cases where an <code>#if</code> is started in one header file and completed in another are diagnosed.
21.1	req		Partial	Only an attempt to define a macro named <code>defined</code> is diagnosed.
21.2	req		No	
21.3	req		No	
21.4	req		Partial	Use of the <code>jmp_buf</code> type is not diagnosed.
21.5	req		Partial	Use of the <code>sig_atomic_t</code> type is not diagnosed. Use of the macros defined in <code>signal.h</code> is not diagnosed.
21.6	req		Partial	Use of the prohibited functions, outside of calling them, is not diagnosed.
21.7	req		Yes	
21.8	req		Yes	
21.9	req		Partial	Use of the prohibited functions, outside of calling them, is not diagnosed.
21.10	req		Partial	Use of the prohibited functions, outside of calling them, is not diagnosed.
21.11	req		Yes	
21.12	adv		Yes	
22.1	req	No	Yes	Rule deemed not decidable but checking is supported.
22.2	mand	No	Partial	Rule is not decidable but assistance is provided.
22.3	req	No	No	
22.4	mand	No	No	
22.5	mand	No	Yes	Rule deemed not decidable but checking is supported.
22.6	mand	No	Yes	Rule deemed not decidable but checking is supported.