

Cross Reference Listing

Cross reference listing of MISRA C:2012 Rules against messages generated

copyright: © 2013 Programming Research

MISRA C:2012 Dir-1.1 (Required)

Any implementation-defined behaviour on which the output of the program depends shall be documented and understood

QAC Messages: 0180, 0202, 0284, 0285, 0286, 0287, 0288, 0289, 0292, 0299, 0320, 0371, 0372, 0375, 0380, 0388, 0390, 0391, 0392, 0410, 0581, 0604, 0609, 0611, 0612, 0614, 0617, 0618, 0634, 0639, 0647, 0715, 0739, 0810, 0828, 0850, 0857, 0858, 0859, 0875, 0930, 1011, 1018, 1027, 1030, 1031, 1053, 1054, 1055, 1056, 2855, 2856, 2857, 2860, 2861, 2862, 2890, 2891, 2892, 2895, 2896, 2897, 3116

Enforcement: NSE - Documentation requirements cannot be statically enforced

MISRA C:2012 Dir-2.1 (Required)

All source files shall compile without any compilation errors

Enforcement: NSE - Compilation process requirements cannot be statically enforced

MISRA C:2012 Dir-3.1 (Required)

All code shall be traceable to documented requirements

Enforcement: NSE - Traceability to documentation requirements cannot be statically enforced

MISRA C:2012 Dir-4.1 (Required)

Run-time failures shall be minimized

QAC Messages: 2791, 2792, 2801, 2802, 2811, 2812, 2821, 2822, 2831, 2832, 2841, 2842, 2845, 2846, 2847, 2871, 2872, 2877

Enforcement: NSE - The requirements are not well-defined and enforcement can never be comprehensive

MISRA C:2012 Dir-4.2 (Advisory)

All usage of assembly language should be documented

QAC Messages: 1003, 1006

Enforcement: NSE - Documentation requirements cannot be statically enforced

MISRA C:2012 Dir-4.3 (Required)

Assembly language shall be encapsulated and isolated

QAC Messages: 3006

Enforcement: NSE - Encapsulation and isolation are not well-defined requirements

MISRA C:2012 Dir-4.4 (Advisory)

Sections of code should not be "commented out"

Enforcement: NSE - The requirement to distinguish between "code" and "comments" is not well-defined

MISRA C:2012 Dir-4.5 (Advisory)

Identifiers in the same name space with overlapping visibility should be typographically unambiguous

Enforcement: NSE - The requirements are only defined for English language environments

MISRA C:2012 Dir-4.6 (Advisory)

typedefs that indicate size and signedness should be used in place of the basic numerical types

QAC Messages: 5209

Enforcement: NSE - The ways in which size and signedness may be specified are not defined

MISRA C:2012 Dir-4.7 (Required)

If a function returns error information, then that error information shall be tested

Enforcement: NSE - The ways in which a function may return error information cannot be defined

MISRA C:2012 Dir-4.8 (Advisory)

If a pointer to a structure or union is never dereferenced within a translation unit, then the implementation of the object should be hidden

Enforcement: NSE - Choosing to use an opaque pointer declaration may not be appropriate in all circumstances

MISRA C:2012 Dir-4.9 (Advisory)

A function should be used in preference to a function-like macro where they are interchangeable

QAC Messages: 3453

Enforcement: NSE - Choosing to use a function rather than a function-like macro is not appropriate in all circumstances

MISRA C:2012 Dir-4.10 (Required)

Precautions shall be taken in order to prevent the contents of a header file being included more than once

QAC Messages: 0883

Enforcement: NSE - The requirement may be met in a variety of ways

MISRA C:2012 Dir-4.11 (Required)

The validity of values passed to library functions shall be checked

Enforcement: NSE - The ways in which function arguments may be checked cannot be defined

MISRA C:2012 Dir-4.12 (Required)

Dynamic memory allocation shall not be used

Enforcement: NSE - The ways in which memory allocation may occur cannot be defined

MISRA C:2012 Dir-4.13 (Advisory)

Functions which are designed to provide operations on a resource should be called in an appropriate sequence

Enforcement: NSE - The requirements are not well-defined

MISRA C:2012 Rule-1.1 (Required)

The program shall contain no violations of the standard C syntax and constraints, and shall not exceed the implementation's translation limits

QAC Messages: 0232, 0233, 0244, 0261, 0321, 0322, 0338, 0422, 0423, 0426, 0427, 0429, 0430, 0431, 0432, 0435, 0436, 0437, 0446, 0447, 0448, 0449, 0450, 0451, 0452, 0453, 0454, 0456, 0457, 0458, 0466, 0467, 0468, 0469, 0476, 0477, 0478, 0481, 0482, 0483, 0484, 0485, 0486, 0487, 0493, 0494, 0495, 0496, 0513, 0514, 0515, 0536, 0537, 0540, 0541, 0542, 0546, 0547, 0550, 0554, 0555, 0556, 0557, 0558, 0559, 0560, 0561, 0562, 0563, 0564, 0565, 0580, 0588, 0589, 0590, 0591, 0605, 0616, 0619, 0620, 0621, 0622, 0627, 0628, 0629, 0631, 0638, 0640, 0641, 0642, 0643, 0644, 0645, 0646, 0649, 0650, 0651, 0653, 0655, 0656, 0657, 0659, 0664, 0665, 0671, 0673, 0674, 0675, 0677, 0682, 0683, 0684, 0685, 0690, 0698, 0699, 0708, 0709, 0736, 0737, 0738, 0746, 0747, 0755, 0756, 0757, 0758, 0766, 0767, 0768, 0774, 0775, 0801, 0802, 0803, 0804, 0811, 0812, 0821, 0834, 0835, 0844, 0845, 0851, 0852, 0856, 0866, 0873, 0877, 0940, 0941, 0943, 0944, 1023, 1024, 1025, 1033, 1047, 1048, 3236, 3237, 3238, 3244

Enforcement: QAC

MISRA C:2012 Rule-1.2 (Advisory)

Language extensions should not be used

QAC Messages: 0240, 0241, 0246, 0551, 0601, 0633, 0635, 0660, 0662, 0830, 0831, 0899, 1001, 1002, 1003, 1006, 1008, 1012, 1014, 1015, 1019, 1020, 1021, 1022, 1026, 1028, 1029, 1034, 1035, 1036, 1037, 1038, 1041, 1042, 1043, 1044, 1045, 1046, 3664

Enforcement: QAC

MISRA C:2012 Rule-1.3 (Required)

There shall be no occurrence of undefined or critical unspecified behaviour

QAC Messages: 0160, 0161, 0162, 0163, 0164, 0165, 0166, 0167, 0168, 0169, 0170, 0171, 0172, 0173, 0174, 0175, 0176, 0177, 0178, 0179, 0184, 0185, 0186, 0190, 0191, 0192, 0193, 0194, 0195, 0196, 0197, 0198, 0199, 0200, 0201, 0203, 0204, 0206, 0207, 0208, 0235, 0275, 0301, 0302, 0304, 0307, 0309, 0337, 0400, 0401, 0402, 0403, 0475, 0543, 0544, 0545, 0602, 0623, 0625, 0626, 0630, 0632, 0636, 0654, 0658, 0661, 0667, 0668, 0672, 0676, 0678, 0680, 0706, 0745, 0777, 0779, 0809, 0813, 0814, 0836, 0837, 0848, 0853, 0854, 0864, 0865, 0867, 0872, 0874, 0885, 0887, 0888, 0914, 0915, 0942, 1331, 1332, 1333, 1509, 1510, 2800, 2810, 2820, 2830, 2840, 3113, 3114, 3239, 3311, 3312, 3319, 3320, 3437, 3438

Enforcement: QAC, Dataflow and CMA

MISRA C:2012 Rule-2.1 (Required)

A project shall not contain unreachable code

QAC Messages: 0594, 1460, 1503, 2742, 2744, 2880, 2882, 3219

Enforcement: QAC, Dataflow and CMA

MISRA C:2012 Rule-2.2 (Required)

There shall be no dead code

QAC Messages: 2980, 2981, 2982, 2983, 2984, 2985, 2986, 2995, 2996, 3110, 3112, 3425, 3426, 3427

Enforcement: QAC and Dataflow

MISRA C:2012 Rule-2.3 (Advisory)

A project should not contain unused type declarations

Enforcement: Not Enforced

MISRA C:2012 Rule-2.4 (Advisory)

A project should not contain unused tag declarations

Enforcement: Not Enforced

MISRA C:2012 Rule-2.5 (Advisory)

A project should not contain unused macro declarations

Enforcement: Not Enforced

MISRA C:2012 Rule-2.6 (Advisory)

A function should not contain unused label declarations

QAC Messages: 3202

Enforcement: QAC

MISRA C:2012 Rule-2.7 (Advisory)

There should be no unused parameters in functions

QAC Messages: 3206

Enforcement: QAC

MISRA C:2012 Rule-3.1 (Required)

The character sequences /* and // shall not be used within a comment.

QAC Messages: 3108, 5133

Enforcement: QAC and Secondary-analysis

MISRA C:2012 Rule-3.2 (Required)

Line-splicing shall not be used in // comments.

QAC Messages: 5134

Enforcement: Secondary-analysis

MISRA C:2012 Rule-4.1 (Required)

Octal and hexadecimal escape sequences shall be terminated

Enforcement: Not Enforced

MISRA C:2012 Rule-4.2 (Advisory)

Trigraphs should not be used

QAC Messages: 3601

Enforcement: QAC

MISRA C:2012 Rule-5.1 (Required)

External identifiers shall be distinct

QAC Messages: 0777

Enforcement: QAC

MISRA C:2012 Rule-5.2 (Required)

Identifiers declared in the same scope and name space shall be distinct

QAC Messages: 0779

Enforcement: QAC

MISRA C:2012 Rule-5.3 (Required)

An identifier declared in an inner scope shall not hide an identifier declared in an outer scope

QAC Messages: 2547, 3334

Enforcement: QAC

MISRA C:2012 Rule-5.4 (Required)

Macro identifiers shall be distinct

Enforcement: Not Enforced

MISRA C:2012 Rule-5.5 (Required)

Identifiers shall be distinct from macro names

Enforcement: Not Enforced

MISRA C:2012 Rule-5.6 (Required)

A typedef name shall be a unique identifier

QAC Messages: 1506, 1507, 1508, 3448

Enforcement: QAC and CMA

MISRA C:2012 Rule-5.7 (Required)

A tag name shall be a unique identifier

Enforcement: Not Enforced

MISRA C:2012 Rule-5.8 (Required)

Identifiers that define objects or functions with external linkage shall be unique

QAC Messages: 1525, 1526

Enforcement: CMA

MISRA C:2012 Rule-5.9 (Advisory)

Identifiers that define objects or functions with internal linkage should be unique

QAC Messages: 1525, 1527, 1528

Enforcement: CMA

MISRA C:2012 Rule-6.1 (Required)

Bit-fields shall only be declared with an appropriate type

QAC Messages: 0634, 0635

Enforcement: QAC

MISRA C:2012 Rule-6.2 (Required)

Single-bit named bit fields shall not be of a signed type

QAC Messages: 3660, 3665

Enforcement: QAC

MISRA C:2012 Rule-7.1 (Required)

Octal constants shall not be used
QAC Messages: 0336, 0339, 3628
Enforcement: QAC

MISRA C:2012 Rule-7.2 (Required)
A "u" or "U" suffix shall be applied to all integer constants that are represented in an unsigned type
QAC Messages: 1281
Enforcement: QAC

MISRA C:2012 Rule-7.3 (Required)
The lowercase character "l" shall not be used in a literal suffix
QAC Messages: 1280
Enforcement: QAC

MISRA C:2012 Rule-7.4 (Required)
A string literal shall not be assigned to an object unless the object's type is "pointer to const-qualified char"
QAC Messages: 0752, 0753
Enforcement: QAC

MISRA C:2012 Rule-8.1 (Required)
Types shall be explicitly specified
QAC Messages: 2050, 2051
Enforcement: QAC

MISRA C:2012 Rule-8.2 (Required)
Function types shall be in prototype form with named parameters
QAC Messages: 1335, 1336, 3001, 3002, 3007
Enforcement: QAC

MISRA C:2012 Rule-8.3 (Required)
All declarations of an object or function shall use the same names and type qualifiers
QAC Messages: 0624, 1330, 3675
Enforcement: QAC

MISRA C:2012 Rule-8.4 (Required)
A compatible declaration shall be visible when an object or function with external linkage is defined
QAC Messages: 3408
Enforcement: QAC

MISRA C:2012 Rule-8.5 (Required)
An external object or function shall be declared once in one and only one file
QAC Messages: 1513, 3221, 3222, 3447, 3451
Enforcement: QAC and CMA

MISRA C:2012 Rule-8.6 (Required)
An identifier with external linkage shall have exactly one external definition
QAC Messages: 0630, 1509, 3406
Enforcement: QAC and CMA

MISRA C:2012 Rule-8.7 (Advisory)
Functions and objects should not be defined with external linkage if they are referenced in only one translation unit
QAC Messages: 1504, 1505
Enforcement: CMA

MISRA C:2012 Rule-8.8 (Required)
The static storage class specifier shall be used in all declarations of objects and functions that have internal linkage
QAC Messages: 3224
Enforcement: QAC

MISRA C:2012 Rule-8.9 (Advisory)
An object should be defined at block scope if its identifier only appears in a single function
QAC Messages: 1514, 3218
Enforcement: QAC and CMA

MISRA C:2012 Rule-8.10 (Required)
An inline function shall be declared with the static storage class
QAC Messages: 3240, 3243
Enforcement: QAC

MISRA C:2012 Rule-8.11 (Advisory)
When an array with external linkage is declared, its size should be explicitly specified
QAC Messages: 3684

Enforcement: QAC

MISRA C:2012 Rule-8.12 (Required)

Within an enumerator list, the value of an implicitly-specified enumeration constant shall be unique

Enforcement: Not Enforced

MISRA C:2012 Rule-8.13 (Advisory)

A pointer should point to a const-qualified type whenever possible

QAC Messages: 3673

Enforcement: QAC

MISRA C:2012 Rule-8.14 (Required)

The restrict type qualifier shall not be used

QAC Messages: 5137

Enforcement: QAC

MISRA C:2012 Rule-9.1 (Mandatory)

The value of an object with automatic storage duration shall not be read before it has been set

QAC Messages: 2883, 2961, 2962, 2971, 2972

Enforcement: Dataflow

MISRA C:2012 Rule-9.2 (Required)

The initializer for an aggregate or union shall be enclosed in braces

QAC Messages: 0693, 0694

Enforcement: QAC

MISRA C:2012 Rule-9.3 (Required)

Arrays shall not be partially initialized

QAC Messages: 0686

Enforcement: QAC

MISRA C:2012 Rule-9.4 (Required)

An element of an object shall not be initialized more than once

Enforcement: Not Enforced

MISRA C:2012 Rule-9.5 (Required)

Where designated initializers are used to initialize an array object the size of the array shall be specified explicitly

Enforcement: Not Enforced

MISRA C:2012 Rule-10.1 (Required)

Operands shall not be of an inappropriate essential type.

QAC Messages: 3101, 3102, 4500, 4501, 4502, 4503, 4504, 4505, 4507, 4510, 4511, 4512, 4513, 4514, 4518, 4519, 4521, 4522, 4523, 4524, 4527, 4528, 4529, 4532, 4533, 4534, 4538, 4539, 4542, 4543, 4544, 4548, 4549, 4558, 4559, 4568, 4569

Enforcement: QAC

MISRA C:2012 Rule-10.2 (Required)

Expressions of essentially character type shall not be used inappropriately in addition and subtraction operations

QAC Messages: 1810, 1811, 1812, 1813

Enforcement: QAC

MISRA C:2012 Rule-10.3 (Required)

The value of an expression shall not be assigned to an object with a narrower essential type or of a different essential type category.

QAC Messages: 0570, 0572, 1257, 1264, 1265, 1266, 1291, 1292, 1293, 1294, 1295, 1296, 1297, 1298, 1299, 2850, 2851, 2852, 2900, 2901, 2902, 4401, 4402, 4403, 4404, 4405, 4410, 4412, 4413, 4414, 4415, 4420, 4421, 4422, 4423, 4424, 4425, 4430, 4431, 4432, 4434, 4435, 4436, 4437, 4440, 4441, 4442, 4443, 4445, 4446, 4447, 4450, 4451, 4452, 4453, 4454, 4460, 4461, 4462, 4463, 4464, 4465

Enforcement: QAC and Dataflow

MISRA C:2012 Rule-10.4 (Required)

Both operands of an operator in which the usual arithmetic conversions are performed shall have the same essential type category

QAC Messages: 1800, 1802, 1803, 1804, 1820, 1821, 1822, 1823, 1824, 1830, 1831, 1832, 1833, 1834, 1840, 1841, 1842, 1843, 1844, 1850, 1851, 1852, 1853, 1854, 1860, 1861, 1862, 1863, 1864, 1880, 1881, 1882

Enforcement: QAC

MISRA C:2012 Rule-10.5 (Advisory)

The value of an expression should not be cast to an inappropriate essential type

QAC Messages: 4301, 4302, 4303, 4304, 4305, 4310, 4312, 4315, 4320, 4322, 4330, 4332, 4340, 4342, 4350, 4351, 4352

Enforcement: QAC

MISRA C:2012 Rule-10.6 (Required)

The value of a composite expression shall not be assigned to an object with wider essential type

QAC Messages: 4490, 4491, 4492, 4499

Enforcement: QAC

MISRA C:2012 Rule-10.7 (Required)

If a composite expression is used as one operand of an operator in which the usual arithmetic conversions are performed then the other operand shall not have wider essential type

QAC Messages: 1890, 1891, 1892, 1893, 1894, 1895, 4397

Enforcement: QAC

MISRA C:2012 Rule-10.8 (Required)

The value of a composite expression shall not be cast to a different essential type category or a wider essential type

QAC Messages: 4390, 4391, 4392, 4393, 4394, 4395, 4398, 4399

Enforcement: QAC

MISRA C:2012 Rule-11.1 (Required)

Conversions shall not be performed between a pointer to a function and any other type

QAC Messages: 0302, 0305, 0307, 0313

Enforcement: QAC

MISRA C:2012 Rule-11.2 (Required)

Conversions shall not be performed between a pointer to an incomplete type and any other type

QAC Messages: 0308

Enforcement: QAC

MISRA C:2012 Rule-11.3 (Required)

A cast shall not be performed between a pointer to object type and a pointer to a different object type

QAC Messages: 0310, 3305

Enforcement: QAC

MISRA C:2012 Rule-11.4 (Advisory)

A conversion should not be performed between a pointer to object and an integer type

QAC Messages: 0303, 0306, 0360, 0361, 0362

Enforcement: QAC

MISRA C:2012 Rule-11.5 (Advisory)

A conversion should not be performed from pointer to void into pointer to object

QAC Messages: 0316, 0317

Enforcement: QAC

MISRA C:2012 Rule-11.6 (Required)

A cast shall not be performed between pointer to void and an arithmetic type

QAC Messages: 0306

Enforcement: QAC

MISRA C:2012 Rule-11.7 (Required)

A cast shall not be performed between pointer to object and a non-integer arithmetic type

QAC Messages: 0301

Enforcement: QAC

MISRA C:2012 Rule-11.8 (Required)

A cast shall not remove any const or volatile qualification from the type pointed to by a pointer

QAC Messages: 0311, 0312

Enforcement: QAC

MISRA C:2012 Rule-11.9 (Required)

The macro NULL shall be the only permitted form of integer null pointer constant

QAC Messages: 3003, 3004

Enforcement: QAC

MISRA C:2012 Rule-12.1 (Advisory)

The precedence of operators within expressions should be made explicit

QAC Messages: 3389, 3391, 3392, 3394, 3395, 3396, 3397

Enforcement: QAC

MISRA C:2012 Rule-12.2 (Required)

The right hand operand of a shift operator shall lie in the range zero to one less than the width in bits of the essential type of the left hand operand

QAC Messages: 0499, 2790

Enforcement: QAC and Dataflow

MISRA C:2012 Rule-12.3 (Advisory)

The comma operator should not be used

QAC Messages: 3417, 3418

Enforcement: QAC

MISRA C:2012 Rule-12.4 (Advisory)

Evaluation of constant expressions should not lead to unsigned integer wrap-around

QAC Messages: 2910

Enforcement: Dataflow

MISRA C:2012 Rule-13.1 (Required)

Initializer lists shall not contain persistent side-effects

Enforcement: Not Enforced

MISRA C:2012 Rule-13.2 (Required)

The value of an expression and its persistent side-effects shall be the same under all permitted evaluation orders

QAC Messages: 0400, 0401, 0402, 0403

Enforcement: QAC

MISRA C:2012 Rule-13.3 (Advisory)

A full expression containing an increment (++) or decrement (--) operator should have no other potential side effects other than that caused by the increment or decrement operator

QAC Messages: 3440

Enforcement: QAC

MISRA C:2012 Rule-13.4 (Advisory)

The result of an assignment operator should not be used

QAC Messages: 3226, 3326

Enforcement: QAC

MISRA C:2012 Rule-13.5 (Required)

The right hand operand of a logical && or || operator shall not contain persistent side effects

QAC Messages: 3415

Enforcement: QAC

MISRA C:2012 Rule-13.6 (Mandatory)

The operand of the sizeof operator shall not contain any expression which has potential side-effects

QAC Messages: 3307

Enforcement: QAC

MISRA C:2012 Rule-14.1 (Required)

A loop counter shall not have essentially floating type

QAC Messages: 3340, 3342

Enforcement: QAC

MISRA C:2012 Rule-14.2 (Required)

A for loop shall be well-formed

QAC Messages: 2461, 2462, 2463, 2464, 2467, 2469, 2471, 2472

Enforcement: QAC

MISRA C:2012 Rule-14.3 (Required)

Controlling expressions shall not be invariant

QAC Messages: 2990, 2991, 2992, 2993, 2994

Enforcement: Dataflow

MISRA C:2012 Rule-14.4 (Required)

The controlling expression of an if-statement and the controlling expression of an iteration-statement shall have essentially Boolean type

QAC Messages: 3344

Enforcement: QAC

MISRA C:2012 Rule-15.1 (Advisory)

The goto statement should not be used

QAC Messages: 2001

Enforcement: QAC

MISRA C:2012 Rule-15.2 (Required)

The goto statement shall jump to a label declared later in the same function

Enforcement: Not Enforced

MISRA C:2012 Rule-15.3 (Required)

Any label referenced by a goto statement shall be declared in the same block, or in any block enclosing the goto statement

QAC Messages: 3311

Enforcement: QAC

MISRA C:2012 Rule-15.4 (Advisory)

There should be no more than one break or goto statement used to terminate any iteration statement

QAC Messages: 0771

Enforcement: QAC

MISRA C:2012 Rule-15.5 (Advisory)

A function should have a single point of exit at the end

QAC Messages: 2889

Enforcement: Dataflow

MISRA C:2012 Rule-15.6 (Required)

The body of an iteration-statement or a selection-statement shall be a compound-statement

QAC Messages: 2212, 2214, 3402

Enforcement: QAC

MISRA C:2012 Rule-15.7 (Required)

All if ... else if constructs shall be terminated with an else statement

QAC Messages: 2004

Enforcement: QAC

MISRA C:2012 Rule-16.1 (Required)

All switch statements shall be well-formed

QAC Messages: 2008, 3234

Enforcement: QAC

MISRA C:2012 Rule-16.2 (Required)

A switch label shall only be used when the most closely-enclosing compound statement is the body of a switch statement

QAC Messages: 2019

Enforcement: QAC

MISRA C:2012 Rule-16.3 (Required)

An unconditional break statement shall terminate every switch-clause

QAC Messages: 2003, 2020

Enforcement: QAC

MISRA C:2012 Rule-16.4 (Required)

Every switch statement shall have a default label

QAC Messages: 2002

Enforcement: QAC

MISRA C:2012 Rule-16.5 (Required)

A default label shall appear as either the first or the last switch label of a switch statement

QAC Messages: 2009

Enforcement: QAC

MISRA C:2012 Rule-16.6 (Required)

Every switch statement shall have at least two switch-clauses

QAC Messages: 3315

Enforcement: QAC

MISRA C:2012 Rule-16.7 (Required)

A switch-expression shall not have essentially Boolean type

QAC Messages: 0735

Enforcement: QAC

MISRA C:2012 Rule-17.1 (Required)

The features of <stdarg.h> shall not be used

QAC Messages: 1337, 5130

Enforcement: QAC and Secondary-analysis.

MISRA C:2012 Rule-17.2 (Required)

Functions shall not call themselves, either directly or indirectly

QAC Messages: 1520, 3670

Enforcement: QAC and CMA

MISRA C:2012 Rule-17.3 (Mandatory)

A function shall not be declared implicitly

QAC Messages: 3335

Enforcement: QAC

MISRA C:2012 Rule-17.4 (Mandatory)

All exit paths from a function with non-void return type shall have an explicit return statement with an expression

QAC Messages: 0745, 2887, 2888, 3113, 3114

Enforcement: QAC and Dataflow

MISRA C:2012 Rule-17.5 (Advisory)

The function argument corresponding to a parameter declared to have an array type shall have an appropriate number of elements

Enforcement: Not Enforced

MISRA C:2012 Rule-17.6 (Mandatory)

The declaration of an array parameter shall not contain the static keyword between the []

Enforcement: Not Enforced

MISRA C:2012 Rule-17.7 (Required)

The value returned by a function having non-void return type shall be used

QAC Messages: 3200

Enforcement: QAC

MISRA C:2012 Rule-17.8 (Advisory)

A function parameter should not be modified

Enforcement: Not Enforced

MISRA C:2012 Rule-18.1 (Required)

A pointer resulting from arithmetic on a pointer operand shall address an element of the same array as that pointer operand

QAC Messages: 2930, 2931, 2932

Enforcement: Dataflow

MISRA C:2012 Rule-18.2 (Required)

Subtraction between pointers shall only be applied to pointers that address elements of the same array

QAC Messages: 2771, 2772

Enforcement: Dataflow

MISRA C:2012 Rule-18.3 (Required)

The relational operators >, >=, < and <= shall not be applied to objects of pointer type except where they point into the same object

QAC Messages: 2771, 2772

Enforcement: Dataflow

MISRA C:2012 Rule-18.4 (Advisory)

The +, -, += and -= operators should not be applied to an expression of pointer type

QAC Messages: 0488

Enforcement: QAC

MISRA C:2012 Rule-18.5 (Advisory)

Declarations should contain no more than two levels of pointer nesting

QAC Messages: 3260, 3261, 3262, 3263

Enforcement: QAC

MISRA C:2012 Rule-18.6 (Required)

The address of an object with automatic storage shall not be copied to another object that persists after the first object has ceased to exist

QAC Messages: 3217, 3225, 3230, 4140

Enforcement: QAC

MISRA C:2012 Rule-18.7 (Required)

Flexible array members shall not be declared

Enforcement: Not Enforced

MISRA C:2012 Rule-18.8 (Required)

Variable-length array types shall not be used

QAC Messages: 0945, 1051, 1052

Enforcement: QAC

MISRA C:2012 Rule-19.1 (Mandatory)

An object shall not be assigned or copied to an overlapping object

QAC Messages: 2776, 2777

Enforcement: Dataflow

MISRA C:2012 Rule-19.2 (Advisory)

The union keyword should not be used

QAC Messages: 0750, 0759

Enforcement: QAC

MISRA C:2012 Rule-20.1 (Advisory)

#include directives should only be preceded by preprocessor directives or comments

QAC Messages: 5087

Enforcement: Secondary-analysis

MISRA C:2012 Rule-20.2 (Required)

The ' , " or \ characters and the /* or // character sequences shall not occur in a header file name

QAC Messages: 0813, 0814, 0831

Enforcement: QAC

MISRA C:2012 Rule-20.3 (Required)

The #include directive shall be followed by either a <filename> or "filename" sequence

QAC Messages: 0809

Enforcement: QAC

MISRA C:2012 Rule-20.4 (Required)

A macro shall not be defined with the same name as a keyword

QAC Messages: 3439

Enforcement: QAC

MISRA C:2012 Rule-20.5 (Advisory)

#undef should not be used

QAC Messages: 0841

Enforcement: QAC

MISRA C:2012 Rule-20.6 (Required)

Tokens that look like a preprocessing directive shall not occur within a macro argument

QAC Messages: 0853

Enforcement: QAC

MISRA C:2012 Rule-20.7 (Required)

Expressions resulting from the expansion of macro parameters shall be enclosed in parentheses

QAC Messages: 3410

Enforcement: QAC

MISRA C:2012 Rule-20.8 (Required)

The controlling expression of a #if or #elif preprocessing directive shall evaluate to 0 or 1

Enforcement: Not Enforced

MISRA C:2012 Rule-20.9 (Required)

All identifiers used in the controlling expression of #if or #elif preprocessing directives shall be #define'd before evaluation

QAC Messages: 3332

Enforcement: QAC

MISRA C:2012 Rule-20.10 (Advisory)

The # and ## preprocessor operators should not be used

QAC Messages: 0341, 0342

Enforcement: QAC

MISRA C:2012 Rule-20.11 (Required)

A macro parameter immediately following a # operator shall not immediately be followed by a ## operator

Enforcement: Not Enforced

MISRA C:2012 Rule-20.12 (Required)

A macro parameter used as an operand to the # or ## operators, which is itself subject to further macro replacement, shall only be used as an operand to these operators

Enforcement: Not Enforced

MISRA C:2012 Rule-20.13 (Required)

A line whose first token is # shall be a valid preprocessing directive

QAC Messages: 3115

Enforcement: QAC

MISRA C:2012 Rule-20.14 (Required)

All #else, #elif and #endif preprocessor directives shall reside in the same file as the #if, #ifdef or #ifndef directive to which they are related

QAC Messages: 3317, 3318

Enforcement: QAC

MISRA C:2012 Rule-21.1 (Required)

#define and #undef shall not be used on a reserved identifier or reserved macro name

QAC Messages: 0836, 0848, 0854, 4600, 4601

Enforcement: QAC

MISRA C:2012 Rule-21.2 (Required)

A reserved identifier or macro name shall not be declared

QAC Messages: 0602, 4602, 4603, 4604, 4605, 4606, 4607, 4608

Enforcement: QAC

MISRA C:2012 Rule-21.3 (Required)

The memory allocation and deallocation functions of <stdlib.h> shall not be used

QAC Messages: 5118

Enforcement: QAC

MISRA C:2012 Rule-21.4 (Required)

The standard header file <setjmp.h> shall not be used

QAC Messages: 5132

Enforcement: Secondary-analysis.

MISRA C:2012 Rule-21.5 (Required)

The standard header file <signal.h> shall not be used

QAC Messages: 5123

Enforcement: Secondary-analysis.

MISRA C:2012 Rule-21.6 (Required)

The Standard Library input/output functions shall not be used

QAC Messages: 5124

Enforcement: Secondary-analysis.

MISRA C:2012 Rule-21.7 (Required)

The atof, atoi, atol and atoll functions of <stdlib.h> shall not be used

QAC Messages: 5125

Enforcement: QAC

MISRA C:2012 Rule-21.8 (Required)

The library functions abort, exit, getenv and system of <stdlib.h> shall not be used

QAC Messages: 5126

Enforcement: QAC

MISRA C:2012 Rule-21.9 (Required)

The library functions bsearch and qsort of <stdlib.h> shall not be used

QAC Messages: 5135

Enforcement: QAC

MISRA C:2012 Rule-21.10 (Required)

The Standard Library time and date functions shall not be used

QAC Messages: 5127

Enforcement: Secondary-analysis

MISRA C:2012 Rule-21.11 (Required)

The standard header file <tgmath.h> shall not be used

QAC Messages: 5131

Enforcement: Secondary-analysis

MISRA C:2012 Rule-21.12 (Advisory)

The exception handling features of <fenv.h> should not be used

QAC Messages: 5136

Enforcement: QAC

MISRA C:2012 Rule-22.1 (Required)

All resources obtained dynamically by means of Standard Library functions shall be explicitly released

Enforcement: Not Enforced

MISRA C:2012 Rule-22.2 (Mandatory)

A block of memory shall only be freed if it was allocated by means of a Standard Library function

Enforcement: Not Enforced

MISRA C:2012 Rule-22.3 (Required)

The same file shall not be open for read and write access at the same time on different streams

Enforcement: Not Enforced

MISRA C:2012 Rule-22.4 (Mandatory)

There shall be no attempt to write to a stream which has been opened as read-only

Enforcement: Not Enforced

MISRA C:2012 Rule-22.5 (Mandatory)

A pointer to a FILE object shall not be dereferenced

Enforcement: Not Enforced

MISRA C:2012 Rule-22.6 (Mandatory)

The value of a pointer to a FILE shall not be used after the associated stream has been closed

Enforcement: Not Enforced