bugSeng

PROFESSIONAL TRAINING COURSES

Effective MISRA C

A professional training course delivered by BUGSENG on MISRA C:2012, the latest version of the MISRA C standard, including the new security amendments. The course has been designed for the smooth and successful adoption of MISRA C into an organization. Lectures, exercises, tests, hands-on sessions and, optionally, a final exam, will significantly strengthen the skills and competences of teams involved in the design, development and verification of critical embedded software systems.

For a number of notorious historical and technical reasons, the C programming language is the most widely used across industry, even in **safety-**, **security- and mission-critical** contexts. Several features of C that proved to be crucial for the success of the language in terms of efficiency and portability, are in sharp conflict with both safety and security requirements. Hence, the development of critical applications requires language subsetting: this is mandated or recommended by all safety- and security-related industrial standards, such as **IEC 61508** (industrial), **ISO 26262** (automotive), **CENELEC EN 50128** (railways), **RTCA DO-178B/C** (aerospace) and **FDA's *General Principles of Software Validation***.

The most authoritative language subset for the C programming language is MISRA C, currently in its third revision, **MISRA C:2012**.

Formal training of personnel involved in the development and quality assessment of C source code is an essential part of the adoption of MISRA C. Without a proper understanding of **C pitfalls** and of the reasons behind each of the MISRA guidelines, developers often:

- perceive the adoption of the guidelines as a useless burden;
- misunderstand messages output by the tool and do not know what should be done;
- are unable to recognize false positives;
- change the code by trial-and-error in an attempt to silence the tools.

Lack of training always implies significant time losses and even, more often than one might think, a strict decrease in the quality of the code produced.

This training course has proven to be **very effective**: it has been used both to successfully introduce MISRA C to companies with no previous exposure to the subject, and to boost productivity in companies that had already adopted MISRA C but without adequate training. In all cases the course resulted in **stronger, more productive teams**.

The course optionally includes **hands-on experience** with ECLAIR, a state-of-the-art software verification platform available from BUGSENG.

Copyright (C) 2010–2018 BUGSENG srl. All other trademarks and copyrights are the property of their respective owners.

Course Objectives

Upon completion of the course, participants will:

- understand the C language pitfalls, the compilation process, static analysis techniques and tools;
- understand the origin and nature of MISRA C and its role in the development of safe and secure software;
- understand all important MISRA C guidelines and the unwanted phenomena they are designed to prevent;
- understand the notion of compliance to MISRA C and the permitted deviation procedures;
- appreciate and understand the advantages of the adoption of MISRA C and other best practices.

In addition, they will be able to:

- recognize and avoid dangerous features of the C language by adhering to the MISRA C language subset, thus minimizing rework and extended testing phases;
- analyze the output of static analyzers and recognize MISRA C false positives (and negatives);
- decide on the best remediation for each kind of MISRA C violation;
- work effectively on bringing projects into compliance;
- formulate accurate and defensible compliance matrices.

Intended Audience and Teaching Methods

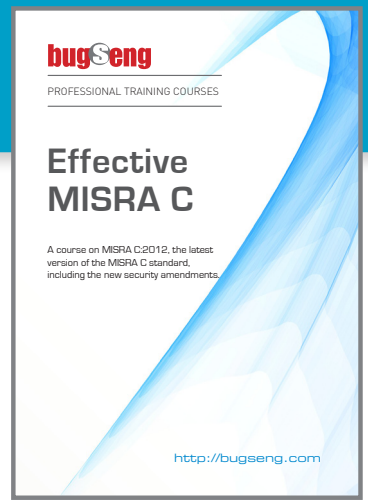
The course is meant for software developers, engineers and architects as well as V&V engineers and project managers.

The content is geared towards people with a working understanding of the C programming language; however, **no previous knowledge of MISRA C is required**.

The course, which favors **participatory approaches** as much as possible, is based on the following methodologies: lectures/presentations, discussions, questions and answers, demonstrations, practical sessions, exercises. An optional final exam can also be provided.

Contents and Schedule

300+ pages of exclusive teaching material



The course provides a thorough understanding of MISRA C, debunking common misconceptions that are usually the reason for its ineffective and counterproductive adoption. The course begins with a presentation of the safety and security pitfalls that are inherent in C programming; the most common and dangerous programming errors (with a particular emphasis on embedded systems programming) are then explained in detail. Each MISRA C rule is presented along with its rationale and the role it plays in achieving safety, testability, maintainability and portability. Most importantly, each rule is presented along with a clear explanation of the right corrective measures (those that do increase overall code quality) and with instructions about why, when and how rule deviations might be necessary or advisable. The use of tools for the automatic verification of MISRA C rules is then introduced, with a particular emphasis on their proper configuration and integration with the development environment.
The course concludes with the demonstrative analysis of a portion of a real software project: this gives the opportunity to review and practice all the learned concepts and abilities.

The course spans two full days. They can be consecutive (recommended) or separated. In the latter case, we recommend that there is no more than two weeks between them.

Copyright (C) 2010–2018 BUGSENG srl. All other trademarks and copyrights are the property of their respective owners.

DAY 1

Morning

1. Introduction to the course.
2. Review of undefined, unspecified and implementation-defined behavior in C.
3. How the compilers may take advantage of undefined behavior.
4. Review of explicit and implicit casts: balancing, promotion, arithmetic conversions, ...
5. Review of enumerated, integer and floating-point types: representation and operations.
6. Review of common integer pitfalls: overflow, sign error, extension, truncation, ...
7. Review of common floating-point pitfalls: error propagation, comparison, excess precision, ...
8. Review of arrays, strings, pointer types and associated programming errors: access outside bounds, null-termination, truncation, off-by-one errors, ...

Afternoon

1. Introduction to MISRA.
2. The purpose of MISRA C and its role in improving code quality.
3. The MISRA C essential type system and other preliminary notions.
4. MISRA C:2012 guidelines related to not fully defined behavior of C.
5. Test on not fully defined behavior of C and related MISRA C guidelines.

Copyright (C) 2010–2018 BUGSENG srl. All other trademarks and copyrights are the property of their respective owners.

DAY 2

Morning

1. Other important MISRA C:2012 guidelines.
2. Security: the new security-oriented guidelines of MISRA C:2012 Amendment 1.
3. Security: review of MISRA C:2012 Addenda 2 and 3 (mapping of MISRA-C:2012 onto ISO/IEC TS 17961:2016 and CERT C:2016).
4. Test on MISRA C violations and the best ways to deal with them.

Afternoon

1. Compliance matrices and deviation procedures: MISRA Compliance:2016.
2. Simplifying the deviation procedure with deviation permits.
3. Automatic verification of compliance to the MISRA C rules: the tools and their proper configuration and use.
4. Demonstrative analysis of the MISRA C violations in a real software project (possibly provided by the customer) along with the correct remediation measures.
5. Final exam (optional) and course wrap-up.

USB key containing an individually-licensed copy of MISRA-C:2012 Revision 1, other MISRA documents, language standards and further reference material

Customization

The course contents can be customized to some extent. For example, in the case of an audience with previous working knowledge of MISRA C (2004 or 2012) the hands-on part of the course can be expanded.

The Instructors

The course is taught by Roberto Bagnara, assisted by other qualified instructors. Roberto is CEO/CTO and Chief Scientist at BUGSENG. He has coauthored more than 40 papers on programming languages, static analysis and other techniques for software verification published in international journals and peer- reviewed conference proceedings. Roberto is a full professor of Computer Science at the University of Parma, where he teaches courses on programming languages and (automated) software verification. He has been working on embedded systems' software since 1984, initially at the University of Bologna (medical devices) and then at CERN (particle physics apparata). He is a member of the MISRA C Working Group and is also a member of the ISO JTC1/SC22/WG14 international standardization working group for the C programming language.

Handouts

Each participant will receive:

- All relevant MISRA documents in PDF format, including a copy of MISRA C:2012 Revision 1 (licensed individually to each participant).
- Printed course material including examples and exercises for individual study (confidential).
- Certificate of achievement.
- One month of free email consultancy on the course topics.

Venues

Standard locations are Parma, Pisa, Milan or Rome depending on requests. The course can also be delivered on-site.

bugseng no shortcuts, no compromises, no excuses: software verification done right
BUGSENG srl • Parco Area delle Scienze 53/A – I-43124 Parma, Italy
Tel: +39 0521 906 906 – Fax: +39 0521 906 950
Email: info@bugseng.com – Web: http://bugseng.com