

**Module: Secure Programming**

<b>Level</b>	Master	<b>Short Name</b>	SPRG
<b>Responsible Lecturers</b>	Oliver Stecklina, Prof. Dr.		
<b>Department, Facility</b>	Electrical Engineering and Computer Science		
<b>Course of Studies</b>	Applied Information Technology, Master		
<b>Compulsory/elective</b>	Elective	<b>ECTS Credit Points</b>	5
<b>Semester of Studies</b>	2	<b>Semester Hours per Week</b>	4
<b>Length (semesters)</b>	1	<b>Workload (hours)</b>	150
<b>Frequency</b>	WiSe	<b>Presence Hours</b>	60
<b>Teaching Language</b>	German/English	<b>Self-Study Hours</b>	90

The following section is filled only if there is **exactly one** module-concluding exam.

<b>Exam Type</b>	Oral Exam	<b>Exam Language</b>	German/English
<b>Exam Length (minutes)</b>	30	<b>Exam Grading System</b>	
<b>Learning Outcomes</b>	After successfully completing the course, students will be able to: <ul style="list-style-type: none"> <li>• Explain and apply methods and technologies for writing safe programs</li> <li>• Develop programs using the Ada programming language</li> <li>• C programs comply with the MISTRA-C standard</li> </ul> Analyze programs with regard to safety-critical program requirements		
<b>Participation Prerequisites</b>			

The previous section is filled only if there is **exactly one** module-concluding exam.

<b>Consideration of Gender and Diversity Issues</b>	<ul style="list-style-type: none"> <li>✓ Use of gender-neutral language (THL standard)</li> <li>✗ Target group specific adjustment of didactic methods</li> <li>✓ Making subject diversity visible (female researchers, cultures etc.)</li> </ul>
<b>Applicability</b>	
<b>Remarks</b>	

## Module Course: Secure programming (Lecture)

(of Module: Secure Programming)

<b>Course Type</b>	Lecture	<b>Form of Learning</b>	Presence
<b>Mandatory Attendance</b>	no	<b>ECTS Credit Points</b>	3
<b>Participation Limit</b>		<b>Semester Hours per Week</b>	3
<b>Group Size</b>		<b>Workload (hours)</b>	90
<b>Teaching Language</b>	German/English	<b>Presence Hours</b>	45
<b>Study Achievements ("Studienleistung", SL)</b>		<b>Self-Study Hours</b>	45
<b>SL Length (minutes)</b>		<b>SL Grading System</b>	

The following section is filled only if there is a course-specific exam.

<b>Exam Type</b>		<b>Exam Language</b>	
<b>Exam Length (minutes)</b>	120	<b>Exam Grading System</b>	
<b>Learning Outcomes</b>			
<b>Participation Prerequisites</b>			

The previous section is filled only if there is a course-specific exam.

<b>Contents</b>	<p>Autonomous and safety-critical systems increasingly determine our everyday life. They influence what we do or how we can do it. Errors in these systems often have a direct impact on us and our daily life. Errors can be made in the design but also in the implementation. However, hardware design must also be carried out with great care, especially when programming applications for safety-critical systems. In order to support the creation of the programs, rules for programming can be set up or secure programming languages can be used.</p> <p>The module "Secure Programming" gives the students an insight into the safe programming of applications for safety-critical (autonomous) systems. On the one hand, rules for unsafe languages, such as C, and on the other hand secure programming languages, e.g. ADA, considered. Practical examples investigate how rules and programming languages help to avoid errors when implementing a safety-critical application.</p> <p>Part 1 - Introduction to Safe Programming Part 1 - MISRA C Part 2 - Programming in ADA and SPARK</p>
<b>Literature</b>	<p>[1] OWASP Secure Coding Practices Quick Reference Guide, 2010</p> <p>[2] John Barnes; Programming in Ada 2012, Cambridge University Press, 2014</p>

[3] John W. McCormick, Building High Integrity Applications with SPARK, Cambridge University Press, 2015

[4] Brian W. Kernighan; The C Programming Language; Prentice Hall 2000

[5] MISRA C: 2012 Guidelines for the use of the C language in critical systems, 2012

<b>Remarks</b>	
----------------	--

## Module Course: Secure Programming (Practical Training)

(of Module: Secure Programming)

<b>Course Type</b>		<b>Form of Learning</b>	Presence
<b>Mandatory Attendance</b>	no	<b>ECTS Credit Points</b>	2
<b>Participation Limit</b>		<b>Semester Hours per Week</b>	1
<b>Group Size</b>	12	<b>Workload (hours)</b>	60
<b>Teaching Language</b>	German/English	<b>Presence Hours</b>	15
<b>Study Achievements ("Studienleistung", SL)</b>	Practical Training	<b>Self-Study Hours</b>	45
<b>SL Length (minutes)</b>		<b>SL Grading System</b>	Pass

The following section is filled only if there is a course-specific exam.

<b>Exam Type</b>		<b>Exam Language</b>	
<b>Exam Length (minutes)</b>		<b>Exam Grading System</b>	
<b>Learning Outcomes</b>			
<b>Participation Prerequisites</b>			

The previous section is filled only if there is a course-specific exam.

<b>Contents</b>	The knowledge of the lecture is to be consolidated in practical examples: <ul style="list-style-type: none"> <li>• Software analysis of sample programs,</li> <li>• Programming according to MISRA-C, and</li> </ul> Programming in Ada.
<b>Literature</b>	
<b>Remarks</b>	