*Only the german version of the module description as part of the study regulations is legally binding.*

| **Module number** | **Module name** | **Responsible lecturer** |
|---|---|---|
| INF-DSE-20-E-CPLCD | Foundations of Certified Programming Language and Compiler Design | Prof. Castrillon jeronimo.castrillon@mailbox.tu-dresden.de |
| **Qualification objectives** | After the successful completion of this module, participants will be able to develop programs with a strong guarantee of their own in order to minimize the testing effort and avoid complex runtime errors during the development process. For this purpose, the participants are familiar with the basics of strongly-typed program languages and their connection to logic. This knowledge enables participants to develop programs in programming languages such as Agda or Theorem Provern such as Coq and to formally prove their properties. The participants are familiar with essential evidence procedures for formally verifying properties of programming languages, compilers and even hardware-related programs. | |
| **contents** | The module conveys the theories of untyped and typed lambda calculus, as well as of type systems with dependent types and their connection to the logic of statements and predicates as a basis for curry-ho ward isomorphism. Incidentally, the module includes programming with strongly-typed programming languages, e.g. Agda, and The- orem Provern, e.g. Coq. On this basis, we discuss the characteristics of programming languages and compilers and the possibilities of formally proving them in the design process of programming languages and compilers. For this purpose, important evidence techniques are introduced and the formal concepts for describing the various semantic definitions (operational, denotational) are conveyed. | |
| **Forms of teaching and learning** | The module includes lectures in the scope of 2 SWS and exercises in the scope of 2 SWS as well as self-study. | |
| **Requirements for participation** | Knowledge and skills on the basics of programming and logic at bachelor level are required. | |
| **usability** | The module is a compulsory elective module of specialisation in the Master's programme Distributed Systems Engineering. | |
| **Conditions for awarding credits** | The credit points are acquired when the module examination is successful. The module examination consists of an oral examination lasting 30 minutes. | |
| **Credits and grades** | 6 credit points can be earned through the module. The Mo- dulnote corresponds to the grade of the examination performance. | |
| **Frequency of the module** | The module is offered every winter semester. | |

| | |
|---|---|
| **workload** | The workload is 180 hours. |
| **Duration of the module** | The module covers one semester. |