



Appendices for the Bachelor's degree programme in Computing Science 2022-2023

- I. Learning outcomes
- II. Majors and Minors
- III. Course units propaedeutic phase
- IV. Course units post-propaedeutic phase
- V. Admission to post-propaedeutic phase
- VI. Contact hours propaedeutic and post-propaedeutic phase
- VII. Additional Requirements Open degree Programmes
- VIII. Transitional provisions



Appendix I Learning outcomes of the Bachelor's degree programme (Article 3.1.1)

Holders of a Bachelor's degree in Computing Science:

1. Have the following knowledge, understanding and insights on an academic level:
 - a. knowledge of the main topics of Computing Science;
 - b. understanding of common themes and principles of Computing Science on different levels of abstraction;
 - c. insight in the applicability of Computing Science and the interplay between theory and practice;
 - d. either d1: in-depth knowledge of certain topics in the field of Computing Science, or d2: broad-based knowledge of topics in a different discipline.

2. Have the following skills and competences, on an academic level:
 - a. technical skills associated with Computing Science, including relevant mathematical and logical skills;
 - b. relevant soft skills, including communication, intercultural teamwork and self-managed learning;
 - c. academic skills, including conceptual thinking, critical questioning, judgement forming, scientific research, writing and presenting in English;
 - d. the competence to analyse, structure, redefine and solve problems, using computational methods and tools;
 - e. the competence to design, develop and evaluate computer systems;
 - f. the competence to apply their knowledge and understanding of Computing Science in a globalized professional and entrepreneurial context.

3. Have the following attitudes:
 - a. appreciation of the role and importance of mathematics, related disciplines and domain-specific knowledge;
 - b. commitment to professional responsibility, including ethical, societal and intercultural issues, with a self-critical attitude;
 - c. critical and academic attitude towards information and knowledge;
 - d. preparedness to life-long learning, based on the awareness of the highly dynamical character of Computing Science.

Appendix II Majors and Minors of the degree programme (Article 3.7.4)

The degree programme has one Major: Computing Science.



Appendix III Course units in the propaedeutic phase

- List of course units; Article 4.1.1
- Compulsory order of examinations; Article 9.3

Course unit name	Course code	ECTS
Imperative Programming	WBCS003-05	5
Introduction to Computing Science	WBCS005-05	5
Introduction to Logic (CS)	WBCS030-05	5
Discrete Structures	WBCS011-05	5
Computer Architecture	WBCS010-05	5
Calculus for Computing Science	WBCS036-05	5
Algorithms and Data Structures in C (for CS)	WBCS018-05	5
Introduction to Information Systems	WBCS021-05	5
Program Correctness	WBCS024-05	5
Advanced Algorithms and Data Structures	WBCS009-05	5
Object-Oriented Programming for CS	WBCS028-05	5
Linear Algebra & Multivariable Calculus	WBAI050-05	5
Total		60

Appendix IV Course units in the post-propaedeutic phase

- List of course units; Article 7.1.1

Course unit name	Course code	ECTS
Functional Programming	WBCS002-05	5
Advanced Object Oriented Programming	WBCS001-05	5
Statistics	WBAI049-05	5
Problem Analysis and Software Design	WBCS012-05	5
Web Engineering	WBCS008-05	5
Signals and Systems for CS	WBCS042-05	5
Introduction to Scientific Computing	WBCS022-05	5
Software Engineering	WBCS017-10	10
Computing Science: Ethical and Professional Issues	WBCS020-05	5
Parallel Computing	WBCS029-05	5
Languages and Machines	WBCS027-05	5
Minor (content determined by the student). Optional elective course units offered by Computing Science that may be used to fill (part of) the minor:		30
-Information Security	WBCS004-05	5
-Network Centric Systems	WBCS031-05	5
-Introduction to Machine Learning	WBCS032-05	5
-Software Language Engineering	WBCS016-05	5
-Short programming project	WBCS015-05	5
-Programming in C++	WBCS034-05	5
-Advanced programming in C++	WBCS035-05	5
-Compiler Construction	WBCS039-05	5
-Information Retrieval	WBCS040-05	5



-Computational Complexity	WBCS044-05	5
Research Skills in Computing Science	WBCS026-05	5
Computer Graphics	WBCS019-05	5
Operating Systems	WBCS023-05	5
Bachelor's project	WBCS901-15	15
Total		120
C++ Fundamentals- <i>course not part of the CS curriculum but elective for other programmes- 5 ECTS</i>	WBCS033-05	-

- **Compulsory order of examinations; Article 9.3**

The examinations for the course units listed below may not be taken before the examinations for the associated course units have been passed:

- Signals and Systems after having passed Calculus for Computing Science and Linear Algebra & Multivariable Calculus.
- Computer Graphics after having passed Calculus for Computing Science and Linear Algebra & Multivariable Calculus.
- Bachelor's project after having completed the propaedeutic phase and earned 80 ECTS from years 2 and 3. Additionally, students will need to have an approved study programme in Progress Portal before they can start this course.
- C++ Fundamentals after having passed Object-Oriented Programming and Imperative Programming or equivalent.
- Programming in C++ after having passed C++ Fundamentals assignments. Alternatively for Computing Science students: after having passed Advanced Object-Oriented Programming.
- Advanced Programming in C++ after having passed Programming in C++ assignments.

Appendix V Admission to the post-propaedeutic phase (Article 6.1.1)

The following candidates will be admitted to the post-propaedeutic phase:
Students who have been issued a positive study advice from the degree programme in question.

Appendix VI Contact hours propaedeutic and post-propaedeutic phase (Article 3.6)

Degree programme year 1	
Structure contact hours	Contact hours per year
Lectures	280
Tutorials	216
Practical	112
Study support/Mentor groups	-
Internship support and guidance	-
Exams	43



Appendix VII Additional Requirements Open degree Programmes (Art. 7.3)

In exceptional circumstances students wishing to pursue an open degree programme may file a request with the Board of Examiners. The Board of Examiners will evaluate whether the proposed curriculum meets the learning outcomes of the degree programme and can determine further conditions in their Rules and Regulations.

Appendix VIII Transitional provisions (article 12.1)

Transitional arrangement for the Bachelor's programme in Computing Science:

Course units			Substitute course units				
Course unit code	Course unit name	EC TS	Course unit code	Course unit name	EC TS	Note	Equivalent* Yes/No
INBTA-08	Languages and Machines	5	INBG AD-10	Advanced Algorithms and Data Structures	5	Students who started in the academic year 2019-2020 switch to the new curriculum with Web Engineering in their major. In year 2, semester IIb, they take Advanced Algorithms and Data Structures, and either Parallel Computing or Languages and Machines (i.e., one of the latter two courses becomes optional).	No
or INBPA R-08	or Parallel Computing						

* It is also possible to substitute equivalent course units in the other direction. This can apply to students with a large backlog who want to fall under the new TER.

Course units			Substitute course units				
Course unit code	Course unit name	EC TS	Course unit code	Course unit name	EC TS	Note	Equivalent* Yes/No
WBCS 18002	Web Engineering	5	WBCS 19000	Research Skills in Computing Science	5	Students who started in the academic year 2018-2019 or earlier can choose to: 1. graduate under the old curriculum (i.e., with Web Engineering as an obligatory course in their major), or 2. switch to the new curriculum (i.e., with CS Research Skills in Computing Science as an obligatory course in their major).	Yes

* It is also possible to substitute equivalent course units in the other direction. This can apply to students with a large backlog who want to fall under the new TER.