



Appendices for the Bachelor's degree programme(s) in Computing Science 2023-2024

- I. Learning outcomes of the Bachelor's degree programme
- 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 1. Learning outcomes of the Bachelor's degree programme (art. 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 Bachelor's degree programme (art. 3.7.4)

The degree programme has one Major: Computing Science.



Appendix III. Course units in the propaedeutic phase

- List of course units (art. 4.1.1)
- Compulsory order of examinations (art. 9.3)

Course unit name	Course code	ECTS
Programming Fundamentals	WBCS046-10	10
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
Object-Oriented Programming (for CS)	WBCS028-05	5
Linear Algebra	WBCS048-05	5
Computer Networks	WBCS047-05	5
Total		60

Appendix IV. Course units in the post-propaedeutic phase

- List of course units (art. 7.1.1)
- Compulsory order of examinations (art. 9.3)

Course unit name	Course code	ECTS
Functional Programming	WBCS002-05	5
Advanced Object Oriented Programming	WBCS001-05	5
Statistics and Probability	WBCS049-05	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
-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- course not part of the CS curriculum but elective for other programmes- 5 ECTS	WBCS033-05	-



As stated in Article 8.2.2, students are free to take any University Minor as a broadening module. The University minor Data Wise: Data Science in Society Minor organized by the Faculty of Behavioural and Social Sciences is not approved, **as the contents significantly overlap with the Computing Science BSc programme.**

It is also not possible to include a course from a University Minor if an equivalent or similar course is already taught in the Computing Science BSc programme.

Formal approval of the Board of Examiners is required, in case and before a student would like to deviate from these rules (e.g. including course units from other programmes and universities).

- **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 (WBCSo42-05) after having passed Calculus for Computing Science (WBCSo36-05) and Linear Algebra & Multivariable Calculus (WBAIo50-05).
- Computer Graphics (WBCSo19-05) after having passed Calculus for Computing Science (WBCSo36-05) and Linear Algebra & Multivariable Calculus (WBAIo50-05).
- Bachelor's project (WBCS901-15) after having completed the propaedeutic phase and earned 75 ECTS from years 2 and 3 of the Bachelor Computing Science. Additionally, students need to have submitted a study programme in Progress Portal. The entry requirements are checked after block 1b.
- C++ Fundamentals (WBCSo33-05) after having passed Object-Oriented Programming (WBCSo28-05) and Imperative Programming (WBCSo03-05) or equivalent.
- Programming in C++ (WBCSo34-05) after having passed C++ Fundamentals (WBCSo33-05) assignments. Alternatively for Computing Science students: after having passed Advanced Object-Oriented Programming (WBCSo01-05).
- Advanced Programming in C++ (WBCSo35-05) after having passed Programming in C++ assignments.



Appendix V. Admission to post-propaedeutic phase (art. 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 (art. 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 (art. 12.1)

The propaedeutic phase of the 2023/24 curriculum has three new compulsory course units (Programming Fundamentals- WBCSo46-10, Linear Algebra -WBCSo48-05 and Computer Networks- WBCSo47-05). Program Correctness- (WBCSo24-05) and Imperative Programming for CS (WBCSo03-05) are discontinued.

Discontinued courses for cohort 2022-2023 and earlier

Discontinued course	Course code	Replacement course	Course code
Imperative Programming (for CS)	WBCSo03-05	Imperative Programming (for AI)	WBAIo03-05
Program Correctness	WBCSo24-05	Compiler Construction <i>or</i> Computational Complexity	WBCSo39-05 <i>or</i> WBCSo44-05
Imperative Programming for CS AND Program Correctness	WBCSo03-05 AND WBCSo24-05	Programming Fundamentals	WBCSo46-10

For cohort 2022-2023 and earlier Linear Algebra & Multivariable Calculus (WBAIo50-05) is compulsory, for cohort 2023-2024 Linear Algebra (WBCSo48-05) is compulsory. Students that have already successfully finished Linear Algebra & Multivariable Calculus (WBAIo50-05) are not allowed to take Linear Algebra (WBCSo48-05).

Program Correctness can be replaced by either Compiler Construction (WBCSo39-05) or Computational Complexity (WBCSo44-05) (i.e. one elective course becomes a compulsory course).

Students that have not successfully finished both Imperative Programming for CS (WBCSo03-05) and Program Correctness (WBCSo24-05) can take Programming Fundamentals (WBCSo46-10) as a replacement course.

Advanced Algorithms and Data Structures (WBCSo09-05) will not be offered in 2023-2024, there will be an exam and resit in 2023-2024. The course will be offered again in 2024-2025.

Students that have already successfully finished Network Centric Systems (WBCSo31-05) are not allowed to take Computer Networks (WBCSo47-05) due to overlap in content.

Discontinued course for cohort 2022-2023

Statistics	WBAIo49-05	Statistics and Probability	WBCSo49-05
------------	------------	----------------------------	------------

For cohort 2022-2023 and later Statistics and Probability (WBCSo49-05) is compulsory. For cohort 2021-2022 and earlier Statistics (WBAIo49-05) is compulsory. Students that have successfully finished Statistics (WBAIo49-05) cannot take Statistics and Probability (WBCSo49-05).



--

Course units			Substitute course units				Equivalent* Yes/No
Course unit code	Course unit name	ECTS	Course unit code	Course unit name	ECTS	Note	
INBTA-08	Languages and Machines	5	INBGAD-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 INBPAR-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.