Assessment Plan bachelor degree programme Computing Science 2022-2023

27 June 2022

Introduction

This Assessment Plan is a systematic description of the way of examination and assessment of all study components in the bachelor degree programme Computing Science 2021-2022 that are offered by the department of Computing Science. It is based on the assessment policy of the Faculty of Science and Engineering (FSE), as set out in the 2016-2017 version of the FSE Quality Assurance Manual (Chapter 6: Assessment policy) and the protocol setting out the duties and powers of the Faculty's Board of Examiners (Manual for Board of Examiners 2017-2018, Chapter 4: Duties of the Board of Examiners).

The Assessment Plan consists of the following parts.

1. Objective of the degree programme.
2. Learning outcomes of the degree programme.
3. An overview of the examiners and the mode of assessment for each study component in the degree programme that is offered by the department Computing Science.
4. A matrix showing the relationship between the study components and the learning outcomes.
5. A description of the curriculum of the degree programme.
6. Descriptions of the content of each study component, consisting of:

    - objectives
    - description of content
    - lecturers
    - assessment mode(s) and calculation of final mark
    - mode(s) of instruction.

Parts 5 and 6 are not incorporated in this document. For part 5 we refer to the sections III, IV and V of the *Appendix of the Teaching and Examination Regulations BSc Computing Science 2021-2022*. Part 6 is available in the online course catalogue Ocasys[1], which provides detailed information for all items mentioned under point 6 above for each of the courses in the Computing Science bachelor programme.

1 http://www.rug.nl/ocasys/

1. Objective of the degree programme

The objective of the bachelor programme Computing Science is to teach the students a combination of *knowledge* (including facts, understanding, and insights), *skills* (including competences) and *attitudes* (including best practices and interaction skills) in Computing Science that both enables them to function professionally on an academic level in Computing Science, and qualifies them for an academic master programme in Computing Science or a related academic discipline.

2. Learning outcomes of the degree programme

Graduates of the Bachelor degree programme Computing Science:

1. Have the following knowledge, understanding and insights on an academic level:

    a. knowledge of the main topics of the Computing Science discipline;
    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 its theory and practice components;
    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 at the bachelor level.

2. Have the following skills and competences, on an academic level:

    a. technical skills associated with Computing Science, including relevant mathematical, analytic, and logical skills;
    b. relevant soft skills, including communication, presentation (oral and in writing), intercultural teamworking, and self-managed learning;
    c. academic skills, including conceptual thinking, critical questioning and argumentation, judgement forming, scientific research, and writing and presenting in English;
    d. the competence to analyze, structure, and (re)define and solve problems, using computational methods and tools;
    e. the competence to design, develop and evaluate (test) 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.

3. Overview of the examiners and a the mode of assessment for each study component offered by the department Computing Science

For the assessment methods, the following abbreviations are used:
WE: written exam
OR: oral exam
TG: participation in tutor group, including presentation
ME: written midterm exam
HW: homework
LA: computer lab assignment
LAR: computer lab assignment, including reporting
OA: other assignments, including reporting
P: presentation
TH: thesis

First year (propaedeutic phase):

course name	1st and 2nd examiner	assessment
Imperative programming	A. Meijster, D. Feitosa	LA, ME, WE
Introduction to Logic	H.H. Hansen, D.Düstegör	HW, ME, WE
Introduction to Computing Science	M.H.F. Wilkinson, S.D. Frey	TG, HW, LAR, WE
Discrete Structures	D.Düstegör , H.H. Hansen	WE
Computer Architecture	G. Gaydadjiev, A. Shahahrami	LAR, WE
Calculus for CS**	P. Tibboel(Math), J. Kosinka	OA, WE, ME
Algorithms and Data Structures in C	S.D. Frey, D. Feitosa	LAR, WE
Program Correctness	J. Perez, D. Düstegör	HW, WE
Introduction to Information Systems	D. Karastoyanova, F.F.M. Mohsen	HW, LAR, WE
Object-Oriented Programming	M.A.M. Soliman, D. Feitosa	LAR, WE
Linear Algebra and Multivariable Calculus**	R.M. Mendez – provided by Mathematics	WE, OA, ME
Advanced Algorithms and Data Structures	K. Bunte, tba	LAR, WE

Second year:

course name	1st and 2nd examiner	assessment
Advanced Object-Oriented Programming	D. Feitosa, R. Smedinga	LAR, WE
Functional Programming	A. Meijster, H. Hansen	LAR, WE
Statistics **	H.A. de Weerd – (AI)	WE, OA
Problem Analysis and Software Design	J.S. van der Ven, R. Smedinga	OA, P
Web Engineering	V. Andrikopoulos, A. Lazovik	LAR, WE
Signals and Systems for CS	Meijster, H. De Weerd (AI)	LAR, WE
Introduction to Scientific Computing	J. Guo, tba	LA, WE
Computing Science: Ethical and Professional Issues	G.A.H. Laugs, M. Riveni	OA, WE, P

Software Engineering (part 1 & 2)	A. Capiluppi, P. Avgeriou	OA, LAR
Languages and Machines	J.A. Perez, D. Frumin	HW, WE
Parallel Computing	G. Gaydadjiev, A. Shahbahrami	WE

Third year:

course name	**1st and 2nd examiner**	**assessment**
Information Security*	F. Turkmen, F.F.M. Mohsen	LAR, WE
Introduction to Machine Learning*	M. Biehl, J. Guo	LAR, WE
Network Centric Systems*	B. Koldehofe, G. Gaydadjiev	WE, HW, LA
Information Retrieval*	G. Azzopardi, M. Riveni	WE, P, OA
Computational Complexity*	D.R.S. Ramanayake, J.A. Perez	OA
Programming in C++*	J. Bokma, D. Feitosa	P, OA
Software Language Engineering*	T. van der Storm, tba	WE
Compiler Construction*	A. Meijster, J.A. Perez	OA, WE
Short Programming Project*	K. Bunte, tba	OA
Operating Systems	F.F.M. Mohsen, A. Shahbahrami	LAR, WE
Computer Graphics	J. Kosinka, S.D. Frey	LAR, WE
Research Skills in Computing Science	J. Kosinka, R. Smedinga	OA
Advanced programming in C++*	J. Bokma, D. Feitosa	P, OA
Bachelor's project	[supervising teachers] P. Avgeriou, G. Azzopardi	TH, P

*elective course
** courses provided by Artificial Intelligence or Mathematics. Their examiners are not listed as BSc Computing Science examiners.

List of examiners for the Bachelor Computing Science 2022-2023:

Name	Examiner Bachelor project	Course examiner
dr. V. Andrikopoulos	X	X
prof. dr. ir. P. Avgeriou	X	X
dr. G. Azzopardi	X	X
prof. dr. M. Biehl	X	X
J. Bokma		X
prof dr. K. Bunte	X	X
prof. dr. A. Capiluppi	X	X
dr. D. Düstegör	X	X
dr. D. Feitosa	X	X
dr. S.D. Frey	X	X
prof. dr. ir. G. Gaydadjiev	X	X
dr. J. Guo	X	X
prof. dr. H.H. Hansen	X	X
prof. dr. D. Karastoyanova	X	X
prof. dr. B. Koldehofe	X	X
prof. dr. J. Kosinka	X	X
G.A.H. Laugs, MSc	X	X
prof. dr. A. Lazovik	X	X
dr. A. Meijster	X	X
dr. F.F.M. Mohsen	X	X
M.Mohammadi, MSc		X
prof. dr. N. Petkov	X	
prof. dr. J.A. Perez Parra	X	X
dr. D.R.S. Ramanayake	X	X
dr. A. Rastogi	X	X
prof. dr. G.R. Renardel de Lavalette	X	
dr. M. Riveni	X	X
prof. dr. J.B.T.M. Roerdink	X	
dr. A. Shahbahrami	X	X
dr. R. Smedinga	X	X
dr. M.A.M. Soliman	X	X
prof. dr. M. Stal	X	
Dr. O.T. Tursun	X	X
prof. dr. T. van der Storm	X	X
dr. F. Turkmen	X	X
J.S. van der Ven, drs		X
dr. M.H.F. Wilkinson	X	X

For thesis projects and the Short Programming Project, PhD students and Postdocs can serve as second examiner.

4. Relation between the study components and the learning outcomes

For completeness, in the table below all courses which are part of the Computer Science bachelor programme are listed. Apart from the courses offered by the department Computer Science (detailed in Section 3), these include also courses offered by other departments (Mathematics and Artificial Intelligence). We include these in the overview below since, this way, a complete picture of the coverage of the learning outcomes is obtained.

The learning outcomes (1a…3d) referred to in the table below have been already detailed in Section 2. The indicators '1' and '2' in the table cells refer respectively to:

- 1 means a *weak* relation between the corresponding learning outcome and study component is present; the respective course addresses only parts (subcomponents) of the respective learning outcome;
- 2 means: a *strong* relation between the corresponding learning outcome and study component is present; the respective course addresses the respective learning outcome to a great extent.

Block	Course name	1a	1b	1c	1d1	1d2		2a	2b	2c	2d	2e	2f		3a	3b	3c	3d
	First year																	
1.1	Imperative Programming for CS			1				2	2	1	2				1			
1.1	Introduction to Computing Science	2	1	1				1	2	1	1				1	2	1	
1.1	Introduction to Logic (CS)	2	1					2			2				1			
1.2	Computer Architecture	1	2	2	1													
1.2	Calculus for Computing Science	1						2			1				2			
1.2	Discrete Structures	2		1				2			1						2	
1.3	Program Correctness			2				2		1	2		1					
1.3	Introduction to Information Systems	2	2	1				1	1			1	1		2			
1.3	Algorithms and Data Structures in C	2	1	1				2	1		2	1					1	
1.4	Advanced Algorithms and Data Structures	2	1	1				2	1	1	1	1			2		1	1
1.4	Linear Algebra & Multivariable Calculus	1						2			1				2			
1.4	Object-Oriented Programming		2	2					1		2	1			1			

Block	Course name	1a	1b	1c	1d1	1d2		2a	2b	2c	2d	2e	2f		3a	3b	3c	3d
	Second year																	
2.1	Advanced Object Oriented Programming			2	2				1		2	1						1
2.1	Functional Programming		1	2				2		1	2				1			
2.1	Statistics	1						2			2				1			
2.2	Web Engineering	1	2		2				1		2							1
2.2	Signals and Systems for CS		1	2				2		1			2					
2.2	Problem Analysis and Software Design	1		1	2			1	2		2	2			1			1
2.3	Introduction to Scientific Computing	1		1		2		1	1	1	2				2		1	
2.3	CS: Ethical and Professional Issues			1					2	1						2		1
2.3+2.4	Software Engineering	1							2			2	2			2		
2.4	Languages and Machines	2	1					2		1	2							
2.4	Parallel Computing	2			2			2			1							1

Block	Course name	1a	1b	1c	1d1	1d2		2a	2b	2c	2d	2e	2f		3a	3b	3c	3d
	Third year																	
3.1	Information Security	1	1	1	2			1			1					1	2	1
3.1	Introduction to Machine Learning	2		2	2			2			1				1			1
3.1	Network Centric Systems	1	1	1	2			1			1	2	2			1	1	1
3.1	Information Retrieval	2	2	1				2	1	1	1	1			1		1	
3.1	Computational Complexity	2	2	1	2			2		2	2				2			
3.2	Software Language Engineering	1	1	1	2			2		1	2		1					
3.2	Short Programming Project			1				2	1	1	1	2			1			
3.2	Programming in C++	1	1	1		1		1	2	1	1	2			1	2		
3.2	Compiler Construction	2	2	2	2			2			2	2			2			
3.3	Advanced programming in C++	1	1	1		1		1	2	1	1	1				1	0	
3.3	Operating Systems	2	2	1	1			1		1	1		2					1
3.3	Computer Graphics			2	2			2										1
3.3	Research Skills in Computing Science								2	2	2						1	
3.4	Bachelor project (15 ECTS)									2	2					1	2	
Na	C++ fundamentals**	1		1				1	1		1							

** C++ fundamentals is not part of the CS programme, it is an elective for other degree programmes.