UNIVERSITAS CAROLINA PRAGENSIS

FACULTAS MATHEMATICAE PHYSICAEQUE DISCIPLINAE

Faculty of Mathematics and Physics

STUDY GUIDE
2019/2020

Bachelor and Master Programmes

Table of Contents

Introduction

Dear Student,

Welcome to the Faculty of Mathematics and Physics at Charles University in Prague. Our faculty offers bachelor's, master's and doctoral degree programmes, given in either Czech or English. This document is dedicated to the bachelor's and master's programmes in English, namely to those leading to

- Bachelor of Computer Science
- Master of Computer Science
- Master of Mathematics.

In this introduction we provide the basic information you will need in order to study at our faculty. For supplementary information we refer to the Code of Study and Examination of Charles University in Prague and to the Rules for Organization of Studies at the Faculty of Mathematics and Physics. The subsequent chapters of this document describe the academic calendar for 2019/20 and the curricula for our programmes.

Academic Life

Duration of Study

The standard period of study for a bachelor's programme is three years and for a master's programme two years. The standard period of study for a degree programme is the period of time in which it is possible to successfully finish the respective degree programme if one follows the recommended course of study. The course of study is concluded with a state final examination and its successful completion leads to the degree of Bachelor of Computer Science (Bc.) for a bachelor's degree programme and to the degree of Master of Computer Science or Master of Mathematics (Mgr.) in a master's degree programme. The maximum period of study in a bachelor's degree programme is six years and in a master's degree programme five years.

As a Charles University student, you also have the possibility of undertaking a period of study at a linked institution in Europe under the Erasmus+ exchange programme. Check the website `https://www.mff.cuni.cz/exchange_programmes`.

Study Sections

Each academic year consists of a winter (October – January) and a summer (February – June) semester. In each semester there are typically 13 weeks of teaching and an examination period of 5 weeks. A study programme is subdivided into sections so that progress and compliance with the conditions for registration for the next study section can be regularly monitored; a study section is typically an academic year, although for students enrolling in a bachelor's programme, the first two study sections correspond to semesters (i.e., the first study section is the winter semester and the second study section is the summer semester). At the end of each study section there is an Annual Evaluation of Study, whose purpose is to establish whether the results of your study hitherto qualify you to register for the next study section. (For those in their first year of a bachelor's programme, the Annual Evaluation comes in two parts, one for each semester.) You are entitled to register for the first study section if you have successfully completed the admissions process. You are entitled to register for later study sections if you meet the requirements of the Annual Evaluation (see below). Registration is a confirmation that you are continuing your study at the faculty.

Degree Plan

Study in a degree programme is guided by a degree plan. Our master's programmes are divided into study branches and our bachelor's programme is divided into specializations. The degree plan specifies the following for each degree programme, for each study branch and, where applicable, for each specialization:

- **Obligatory courses** (you have to complete these before the state final examination),
- **Elective courses** (you have to complete the prescribed part of these before the state final examination),
- **State final examination** - its parts and requisite knowledge needed for them,
- **Recommended course of study** (for some programmes) - assignment of obligatory and some elective courses to specific study sections; in some cases also provision of supplementary information on the curriculum.

The recommended course of study is not binding. However it is advisable to follow it because it is put together to satisfy the requisites (see below), considers the relationships between the courses, takes into account the schedule, and leads to timely graduation. All courses other than the obligatory and elective that are offered at the university are considered as **Optional courses** for the corresponding curriculum; it is up to you whether you decide to take some of these.

In all tables, obligatory courses are printed **in boldface**, elective courses are printed upright, and optional courses *in italics*. Here is a small example:

Code	Subject	Credits	Winter	Summer
NPRG030	**Programming I**	6	3/2 C	—
NDMI012	Combinatorics and Graph Theory II	6	2/2 C+Ex	—
NMAI069	*Mathematical skills*	2	0/2 C	—

The course code is given in the first column. The number in the "Credit" column specifies the number of ECTS credits for completion of the course. The Winter and Summer columns specify the semester in which the course is offered, the number of hours of lectures/ hours of classes per week, and how the course is assessed (i.e., by a course credit – C, by an exam – Ex). Please be aware that some elective courses are not taught every year.

Course Enrolment

At the beginning of each semester there is a period of several weeks during which you should choose from and enrol in courses that you plan to take that semester (see the Academic Calendar). Enrolment is performed electronically through the Student Information System (SIS) – `http://www.mff.cuni.cz/sis`; further technical details about course enrolment are provided on the webpage `https://www.mff.cuni.cz/course_enrolment`.

The period for course enrolment is split into two phases: in the first phase (priority mode), you have the right to enrol in courses that are primarily designated for you (e.g., the obligatory courses); in the second phase (open mode), you can enrol in any courses. It is up to you which courses to enrol in, subject to the requirements of your curriculum and to the number of credits required in the Annual Evaluation. If your interests are wider than specified by your curriculum or if you decide not to follow the recommended course of study exactly, then you can enrol in additional courses; there is no upper limit to the number of courses in which you can enrol. Course enrolment may be restricted by certain conditions (requisites), of which the most common are the following:

- **Prerequisite** – A prerequisite to Course X is a course that must be successfully completed before you can enrol in Course X.
- **Corequisite** – A corequisite to Course X is a course that you have to enrol in at the same time as Course X, or that you have already successfully completed.
- **Prohibited combination** (or incompatibility) – Courses X and Y are a prohibited combination if it is impossible to enrol in Course X when you have already completed, or you enrol in, Course Y.

In some cases, it is specified that completion of Course Y is equivalent, with respect to the requirements of the curriculum, to completion of Course X; these two courses are called **equivalent** or interchangeable. Information about these relationships among courses are described in the Student Information System in the module "Subjects" (`http://www.mff.cuni.cz/courses`) and in the List of Courses of the Faculty of Mathematics and Physics (electronic edition). Please note that the prerequisites and corequisites for a course X, as specified in SIS and in the List of Courses, apply to study programmes and study branches in which the course is compulsory or elective. We recommend giving due attention to these conditions, as missing a course that is a prerequisite for another course in which you intend to enrol may result in an unfavourable extension of your period of study.

Lectures and Classes

Courses are mostly given in the form of lectures and/or classes. A lecture is an oral presentation intended to teach students a particular subject. Typically accompanying a series of lectures are classes, in which a tutor helps a small group of students assimilate material from lectures and is able to give students individual attention. Classes for programming-related courses typically take place in computer labs. The schedule of the faculty is given as 45-minute periods with 5-minute breaks, and most lectures and classes are organized as 90-minute long blocks of two such periods, taking place once or twice a week. Attendance of lectures and classes is usually not required, but is strongly recommended. Information about course locations and times is available in the Student Information System in the module "Schedule" (`http://www.mff.cuni.cz/schedule`). Lectures are mostly given by senior faculty members, whose academic ranks are professor (in Czech *profesor*) or associate professor (*docent*). Classes are usually conducted by junior researchers and Ph.D. students. Apart from lectures and classes, another significant component of a student's timetable is private study.

Exams and Course Credits

Mastery of a course is confirmed by a course credit and/or by an exam. A course credit (usually for classes) is awarded at the end of the semester. The conditions for obtaining a course credit differ according to the nature of the course, for example involving the completion of a test, programming an application, or writing a survey, and are specified by the teacher at the beginning of the semester. The possible outcomes are Pass (in Czech *Započteno - Z*) and Fail (*Nezapočteno - K*). Exams are taken during the examination period at the end of the semester and may be oral, written, or a combination of the two. Examination dates are announced by the lecturer at the beginning of the examination period. There are four possible outcomes for an exam (the corresponding numerical values and Czech equivalents are given in parentheses): Excellent (1 - *Výborně*), Very good (2 - *Velmi dobře*), Good (3 - *Dobře*), Fail (4 - *Nevyhověl*). You pass an exam if you obtain a grade of Excellent, Very good or Good; otherwise you fail. You have up to three attempts to pass an exam (provided there are still dates available) but we strongly recommend preparing as well as you can for the first attempt. If you do not succeed in passing the exam or obtaining the course credit for a course, you are allowed to take the course again in the next section of study, but a course can be followed at most twice. For each successfully completed course you obtain a certain number of ECTS credits that is specified for each subject in the curriculum (and also given in SIS and the List of Courses of the Faculty of Mathematics and Physics).

Annual Evaluation of Study

Progress is monitored at the end of each study section. The Annual Evaluation of Study involves a check of your credit total, that is, the number of credits obtained in all previous study sections by the end of the last examination period. If you in previous study sections have attained in total at least the normal number of credits (corresponding to the sum of the credits in these sections in the recommended course of study), or if you have obtained at least the minimum number of credits (see below), then you have the right to enrol in the next study section. Please note that while the Annual Evaluation of Study may come after the end of the official examination period for the previous study section (see the Academic Calendar for the exact dates of the winter and summer examination periods), only credits obtained by the end of the official examination period will be considered as part of the assessment. Attaining at least the normal number of credits is one of the necessary conditions for obtaining a scholarship for excellent study achievement. If you have not received the minimum number of credits, then this is considered as a failure to fulfil the requirements of the study programme and results in exclusion.

The normal and minimum numbers of credits required for registration in the next study section are given as follows (normal number of credits is followed in parentheses by minimum number of credits):

- **Normal and minimum number of credits**
  Bachelor's degree programmes
  - 30 (12) for enrolment to the second study section (i.e., the summer semester of the first year of study),
  - 60 (45) for enrolment to the third study section (i.e., the second year),
  - 120 (90) for enrolment to the fourth study section (i.e., the third year),
  - 180 (135) for enrolment to the fifth study section (i.e., the fourth year),

  - 240 (180) for enrolment to the sixth study section (i.e., the fifth year),
  - 300 (225) for enrolment to the seventh study section (i.e., the sixth year).

Master's degree programmes
  - 60 (45) for enrolment to the second study section (i.e., the second year),
  - 120 (90) for enrolment to the third study section (i.e., the third year),
  - 180 (135) for enrolment to the fourth study section (i.e., the fourth year),
  - 240 (180) for enrolment to the fifth section study (i.e., the fifth year).

For the purpose of the Annual Evaluation of Study, all the credits for completed compulsory and elective courses are counted, and credits for optional courses are counted up to the following limits (in parentheses we specify what percentage of the corresponding normal number of credits the maximum number corresponds to):

- **Maximum number of credits for optional courses**
  Bachelor's degree programmes
  - 4 credits (15 %) for enrolment to the second study section,
  - 9 credits (15 %) for enrolment to the third study section,
  - 18 credits (15 %) for enrolment to the fourth study section,
  - 54 credits (30 %) for enrolment to the fifth study section,
  - 72 credits (30 %) for enrolment to the sixth study section,
  - 90 credits (30 %) for enrolment to the seventh study section.

  Master's degree programmes
  - 18 credits (30 %) for enrolment to the second study section,
  - 60 credits (50 %) for enrolment to the third study section,
  - 126 credits (70 %) for enrolment to the fourth study section,
  - 167 credits (70 %) for enrolment to the fifth section study.

You are allowed to enrol in and complete optional courses with a larger number of credits than the maximum specified above, but then some of these credits will not be considered in the Annual Evaluation of Study. Although you technically only need the minimum number of credits to register for the next study section, we strongly recommend attaining the normal number of credits, otherwise you most likely will not be able to complete your study programme within the standard period of time. Technical details about the Annual Evaluation are provided on the webpages `https://www.mff.cuni.cz/first_annual_evaluation` and `https://www.mff.cuni.cz/annual_evaluation`.

State Final Exam

Studies are concluded with a state final examination. This examination consists of several parts (two or three, depending on the corresponding curriculum), one of which for bachelor's degree programmes is always the defence of a bachelor's thesis and for master's degree programmes the defence of a master's (diploma) thesis. If a student fails a part of the state final exam, only the failed part is repeated. Each part of the state final exam may be repeated at most twice. Each part of the state final exam is graded separately and from these an overall grade is awarded. Necessary conditions for taking the State Final Exam include passing all obligatory courses, obtaining the required number of credits for elective courses, reaching a total of at least 180 credits (in bachelor's degree programmes) or 120 credits (in master's degree programmes), and submitting a completed thesis (for the thesis defence). The knowledge requirements for the State Final Exam are specified in the degree plans of the respective study programmes and branches of study, which are described in this document.

More detailed instructions and advice on the assignment, writing up, submission and defence of bachelor's and master's theses are provided at the webpage `https://www.mff.cuni.cz/final_thesis`.

Some Suggestions

Advising others is always a bit tricky but nevertheless I would like to give here a few suggestions for making the most of your time at our faculty. They are addressed primarily to students in their first year.

Ask questions. As the wise old saying goes, *"Many things are lost for want of asking."* Please do no hesitate to ask when you do not understand something – in our culture it is not considered impolite or improper to do so. Ask the lecturer during the lecture or after, ask the tutor during the class or after it, ask your mentor, ask your classmates who (think that they) understand. Arrange a meeting with your teacher during office hours and ask there.

Write. Many of you will learn faster and better if listening and reading is complemented by writing. These days, for many courses there are excellent written materials, often including slides prepared by the lecturer. Nevertheless, many of you will profit by taking your own notes at lectures and classes. It is important to use pen and paper during your preparation for tests and exams. Do you think that you already understand the proof of a theorem? Write it down with the book closed, making sure to include all necessary details. Finally, if you want to learn to program well, write some code.

Work. Even if nobody makes you do so. In contrast to many secondary schools, you will not have to take a test or do homework every day. However, there will be plenty of tests and exams at the end of the semester. Be aware of this and learn as much as you can during the semester rather than later. You will learn more, you will retain it longer, and the examination period will go more smoothly for you.

Plan. This is related to the previous point. In the examination period you will rarely be able to learn well for an exam during a single day or night. Take this into account when planning the dates of your exams in the examination period. Allow yourself enough time to prepare for exams, to code programs or to solve problems for obtaining a course credit. Reserve some time for possible second attempts at failed exams. Do not postpone until the next semester or the next year what you are to do now. If you do so, most likely you will not be able to catch up.

Think. Not everything that you read on the internet is correct. Not even everything that you hear in a lecture is always correct (we all make mistakes). Try to understand everything. Do not be content merely with answers to the questions how? and what?, but also ask *why?* If you have a question, try first to find an answer by yourself before searching for an answer in a textbook or on the internet.

Best wishes for an enjoyable and successful academic year.

Petr Kolman
Coordinator for Studies in English

Prague, July 12, 2019

Academic calendar

Sep 2 – Sep 13, 2019	Autumn period for bachelor's state final examinations
Sep 4 – Sep 17, 2019	Autumn period for master's state final examinations
Sep 9 – Sep 22, 2019	Electronic enrolment in winter semester courses – priority mode
Sep 23 – Oct 13, 2019	Electronic enrolment in winter semester courses – open mode
until Sep 30, 2019	Annual evaluation for academic year 2018/2019 and registration for second and higher years of bachelor's and master's programmes
Oct 1, 2019	Beginning of academic year 2019/2020 and of its winter semester
Oct 1, 2019 – Jan 12, 2020	Winter semester tuition
Oct 14 – Oct 25, 2019	Approval of electronic enrolment in courses by the Department of Student Affairs
Oct 29, 2019	Matriculation of first year students on bachelor's and master's study programmes
until Nov 1, 2019	Recommended period for deciding bachelor's thesis topics
Nov 12, 2019	Dean's Sports Day (no lectures or classes)
Nov 12 – Nov 13, 2019	Graduation Ceremony – bachelor's study programmes
Nov 21, 2019	Open Day
Dec 5 – Dec 6, 2019	Graduation Ceremony – master's study programmes
Dec 21, 2019 – Jan 3, 2020	Christmas vacation
until Jan 6, 2020	Submission of bachelor's and master's (diploma) thesis for winter period of state final examinations - electronic version
until Jan 7, 2020	Submission of bachelor's and master's (diploma) thesis for winter period of state final examinations - paper version
Jan 13 – Feb 16, 2020	Winter semester examination period
until Jan 17, 2020	Checking compliance with all conditions the final year of bachelor's and master's for admission to the winter term of state final examinations
	Registration for winter period of bachelor's and master's state final examinations

Feb 3 – Feb 14, 2020	Winter period bachelor's and master's state final examinations
Feb 3 – Feb 9, 2020	Electronic enrolment in summer semester courses – priority mode
Feb 10 – Mar 8, 2020	Electronic enrolment in summer semester courses – open mode
until Feb 14, 2020	Recommended period for deciding master's thesis topics
Feb 17 – May 24, 2020	Summer semester tuition
until Feb 28, 2020	For first year bachelor's students: Annual evaluation after the winter semester
Mar 9 – Mar 20, 2020	Approval of electronic enrolment in courses by the Department of Student Affairs
Apr 21, 2020	Graduation Ceremony – master's study programmes
May 6, 2020	Rector's Day (no lectures or classes)
until May 7, 2020	Submission of master's thesis for summer period of state final examinations - electronic version
until May 11, 2020	Submission of master's thesis for summer period of state final examinations - paper version
until May 14, 2020	Submission of bachelor's thesis for summer period of state final examinations - electronic version
until May 18, 2020	Submission of bachelor's thesis for summer period of state final examinations - paper version
until May 25, 2020	Checking compliance with all conditions the final year of master's studies for admission to the summer term of state final examinations
	Registration for summer period of master's state final examinations
May 25 – Jun 30, 2020	Summer semester examination period
until Jun 7, 2020	Checking compliance with all conditions the final year of bachelors studies for admission to the summer term of state final examinations
	Registration for summer period of bachelor's state final examinations
Jun 8 – Jun 19, 2020	Summer period for master's state final examinations
Jun 15 – Jun 26, 2020	Summer period for bachelor's state final examinations
Jul 1 – Aug 31, 2020	Summer vacation
until Jul 23, 2020	Submission of bachelor's and master's (diploma) thesis for autumn period of state final examinations - electronic version

until Jul 24, 2020	Checking compliance with all conditions the final year of bachelor's and master's studies for admission to the autumn term of state final examinations
	Registration for autumn period of bachelor's and master's state final examinations
until Jul 27, 2020	Submission of bachelor's and master's (diploma) thesis for autumn period of state final examinations - paper version
Sep 1 – Sep 11, 2020	Autumn period for bachelor's state final examinations
Sep 2 – Sep 15, 2020	Autumn period for master's state final examinations
Sep 21 – Sep 25, 2020	Examination period
until Sep 30, 2020	Annual evaluation for academic year 2019/2020 and registration for second and higher years of bachelor's and master's study programmes
Sep 30, 2020	End of academic year 2019/2020

Location of faculty buildings

The Faculty of Mathematics and Physics comprises the School of Mathematics, the School of Physics, and the School of Computer Science. The schools are based at several locations in Prague. Here we provide basic information about their locations and about lecture rooms in the corresponding buildings. More details about the internal structure of the Faculty of Mathematics and Physics are given at `http://www.mff.cuni.cz/to.en/fakulta/struktura/`, and more details about locations and directions for faculty buildings are given at `http://www.mff.cuni.cz/to.en/fakulta/budovy/`.

School of Mathematics

Address: Sokolovská 83, 186 00 Praha 8

Lecture rooms

K1, K2, K3, K4, K5, K6, K7, K8, K9, K11, K12

Computer labs

K10

School of Computer Science

Address: Malostranské nám. 25, 118 00 Praha 1

Lecture rooms

S1, S3, S4, S5, S6, S7, S8, S9, S10, S11

Computer labs

SW1, SW2

School of Physics

The School of Physics is based in two locations: V Holešoviškách and Ke Karlovu.

Address: V Holešovičkách 2, 180 00 Praha 8

Lecture rooms
: T1, T2, T5, T6, T7, T8, T9, T10, T11

Computer labs
: LabTF, LabTS

There are two neighbouring faculty buildings in the street Ke Karlovu.

Address: Ke Karlovu 3, 121 16 Praha 2

Lecture rooms
: M1, M2, M3, M5, M6

Computer labs
: PLK

Address: Ke Karlovu 5, 121 16 Praha 2

Lecture rooms
: F1, F2

Charles University Sports Centre

Address: Bruslařská 10, 102 00 Praha 10

Administration

Charles University in Prague

Address: Ovocný trh 5, 116 36 Praha 1

Rector:	prof. MUDr. Tomáš Zima, DrSc., MBA

Faculty of Mathematics and Physics

Address: Ke Karlovu 3, 121 16 Praha 2, phone 221 911 289, fax 221 911 292,
e-mail: sdek@dekanat.mff.cuni.cz

Dean

prof. RNDr. Jan Kratochvíl, CSc.

Advisory Board

Deputy Dean and Vice Dean for Research and International Affairs:	prof. RNDr. Jan Trlifaj, CSc., DSc.
Vice Dean for Student Affairs:	doc. RNDr. František Chmelík, CSc.
Vice Dean for Education:	doc. RNDr. Vladislav Kuboň, Ph.D.
Vice Dean for Development:	prof. RNDr. Ladislav Skrbek, DrSc.
Vice Dean for Physics:	prof. RNDr. Vladimír Baumruk, DrSc.
Vice Dean for Computer Science:	prof. RNDr. Jiří Sgall, DrSc.
Vice Dean for Mathematics:	doc. RNDr. Mirko Rokyta, CSc.
Vice Dean for Public Relations:	doc. RNDr. Martin Vlach, Ph.D.
Coordinator for Studies in English:	doc. Mgr. Petr Kolman, Ph.D.
OP VVV Coordinator:	doc. RNDr. Ctirad Matyska, DrSc.
IT Coordinator:	doc. RNDr. Petr Hnětynka, Ph.D.
Secretary:	Ing. Antonín Líska

Master of Mathematics

1 General Information

Programme coordinator: doc. Mgr. Petr Kaplický, Ph.D.

The study programme Master of Mathematics consists of the following study branches:

- Mathematical Structures
- Mathematics for Information Technologies
- Mathematical Analysis
- Numerical and Computational Mathematics
- Mathematical Modelling in Physics and Technology
- Probability, Mathematical Statistics and Econometrics
- Financial and Insurance Mathematics

Assumed knowledge

Individual branches have specific entry requirements for the knowledge assumed to have been already acquired before the start of the Master's programme. Upon evaluation of the previous study experience of each incoming student, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

State Final Exam

Study in the master's programme is completed by passing the state final exam. It consists of two parts: defence of the master's (diploma) thesis, and an oral examination. Requirements for the oral part of the state final exam are listed in the following sections.

Students are advised to select the topic of their master's (diploma) thesis during the first year of the study. The departments of the faculty offer many topics for master theses each year and students can also suggest their own topics. We recommend to select the topic of your thesis primarily from the offer of the department that coordinates your study branch. If you prefer a topic offered by another department or your own topic, please consult it with the coordinator of your study branch. Work on the master's thesis is recognized by credits awarded upon taking the following courses

Code	Subject	Credits	Winter	Summer
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

A student should enrol in these courses according to instructions of their thesis advisor (after the thesis topic has been assigned). These courses can be taken between the second semester of the first year and the last semester of study, in an arbitrary order and in an arbitrary semester. The credits for these courses are assigned by the thesis advisor. The last credits for these courses should be awarded when the master's thesis has been almost completed.

The deadlines for the assignment of the master's thesis topic, submission of the completed thesis, and enrolment in the final exam are determined by the academic calendar.

Project

A student can request an assignment of a project from the dean. The number of credits earned upon completion of the project (max. 9) is specified by the dean on the recommendation of the project advisor and the coordinator of the study programme.

2 Degree Plans - Mathematical Structures

Coordinated by: Department of Algebra
Study branch coordinator: prof. RNDr. Jan Krajíček, DrSc.

The curriculum is focused on extending general mathematical background (algebraic geometry and topology, Riemann geometry, universal algebra and model theory) and obtaining deeper knowledge in selected topics of algebra, geometry, logic, and combinatorics. The aim is to provide sufficient general knowledge of modern structural mathematics and to bring students up to the threshold of independent research activity. Emphasis is laid on topics taught by instructors who have achieved worldwide recognition in their field of research.

A graduate has advanced knowledge in algebra, geometry, combinatorics and logic. He/she is in close contact with the latest results of contemporary research in the selected field. The abstract approach, extensiveness and intensiveness of the programme result in the development of the ability to analyse, structure and solve complex and difficult problems. Graduates may pursue an academic career or realize themselves in jobs that involve mastering new knowledge and control of complex systems.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Linear algebra, real and complex analysis, and probability theory.
- Foundations of group theory (Sylow theorems, free groups, nilpotence). Lie groups, analysis on manifolds, ring and module theory (finiteness conditions, projective and injective modules), commutative algebra (Galois theory, integral extensions).
- Intermediate knowledge of mathematical logic (propositional and first order logic, incompleteness and undecidability).

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

2.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NMAG401	**Algebraic Geometry**	5	2/2 C+Ex	—
NMAG403	**Combinatorics**	5	2/2 C+Ex	—
NMAG405	**Universal Algebra 1**	5	2/2 C+Ex	—
NMAG407	**Model Theory**	3	2/0 Ex	—
NMAG409	**Algebraic Topology 1**	5	2/2 C+Ex	—
NMAG411	**Riemannian Geometry 1**	5	2/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

2.2 Elective Courses

It is required to earn at least 35 credits from the following elective courses.

Code	Subject	Credits	Winter	Summer
NMAG462	Modular forms and L-functions I	3	2/0 Ex	—
NMAG473	Modular forms and L-functions II	3	—	2/0 Ex
NMAG455	Quadratic forms and class fields I	3	2/0 Ex	—
NMAG456	Quadratic forms and class fields II	3	—	2/0 Ex
NMAG431	Combinatorial Group Theory 1	1	2/0 C	—
NMAG432	Combinatorial Group Theory 2	5	—	2/0 Ex
NMAG433	Riemann Surfaces	3	2/0 Ex	—
NMAG434	Categories of Modules and Homological Algebra	6	—	3/1 C+Ex
NMAG435	Lattice Theory 1	3	2/0 Ex	—
NMAG436	Curves and Function Fields	6	—	4/0 Ex
NMAG437	Seminar on Differential Geometry	3	0/2 C	0/2 C
NMAG438	Group Representations 1	5	—	2/2 C+Ex
NMAG440	Binary Systems	3	—	2/0 Ex
NMAG442	Representation Theory of Finite-Dimensional Algebras	6	—	3/1 C+Ex
NMAG444	Combinatorics on Words	3	—	2/0 Ex
NMAG446	Logic and Complexity	3	—	2/0 Ex
NMAG448	Invariant Theory	5	—	2/2 C+Ex
NMAG450	Universal Algebra 2	4	—	2/1 C+Ex
NMAG452	Introduction to Differential Topology	3	—	2/0 Ex
NMAG454	Fibre Spaces and Gauge Fields	6	—	3/1 C+Ex
NMAG531	Approximations of Modules	3	2/0 Ex	—
NMAG532	Algebraic Topology 2	5	—	2/2 C+Ex
NMAG533	Harmonic Analysis 1	6	3/1 C+Ex	—
NMAG534	Harmonic Analysis 2	6	—	3/1 C+Ex
NMAG536	Proof Complexity and the P vs. NP Problem	3	—	2/0 Ex
NMMB401	Automata and Convolutional Codes	6	3/1 C+Ex	—

NDMI013	Combinatorial and Computational Geometry II	6	—	2/2 C+Ex
NDMI028	Linear Algebra Applications in Combinatorics	6	2/2 C+Ex	—
NDMI045	Analytic and Combinatorial Number Theory	3	—	2/0 Ex
NDMI073	Combinatorics and Graph Theory III	6	2/2 C+Ex	—
NTIN022	Probabilistic Techniques	6	2/2 C+Ex	—
NTIN090	Introduction to Complexity and Computability	5	2/1 C+Ex	—

2.3 State Final Exam

Requirements for taking the final exam

- Earning at least 120 credits during the course of the study.
- Completion of all obligatory courses prescribed by the study plan.
- Earning at least 35 credits by completion of elective courses.
- Submission of a completed Master's Thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of a common subject area "1. Mathematical Structures" and a choice of one of four subject areas "2A. Geometry", "2B. Representation Theory", "2C. General and Combinatorial Algebra", or "2D. Combinatorics". One question is asked from subject area 1 and one question is asked from the subject area selected from among 2A, 2B, 2C, or 2D.

Requirements for the oral part of the final exam

Common requirements

1. Mathematical Structures

Basics of algebraic geometry, universal algebra, Riemannian geometry, algebraic topology, model theory and combinatorics.

Specialization

2A. Geometry

Harmonic analysis and invariants of classical groups, Riemannian surfaces, algebraic topology, fibre spaces and covariant derivation.

2B. Representation Theory

Representations of groups, representations of finite-dimensional algebras. combinatorial group theory, curves and function fields, and homological algebra.

2C. General and Combinatorial Algebra

Finite groups and their representations, combinatorial group theory, binary systems (semigroups, quasigroups), advanced universal algebra (lattices, clones, Malcev conditions), complexity and enumerabilty, undecidability in algebraic systems.

2D. Combinatorics

Applications of linear algebra. combinatorics and graph theory, application of probabilistic method in combinatorics and graph theory, analytic and combinatorial number theory, combinatorial and computational geometry, structural and algorithmic graph theory.

2.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMAG401	**Algebraic Geometry**	5	2/2 C+Ex	—
NMAG403	**Combinatorics**	5	2/2 C+Ex	—
NMAG405	**Universal Algebra 1**	5	2/2 C+Ex	—
NMAG409	**Algebraic Topology 1**	5	2/2 C+Ex	—
NMAG411	**Riemannian Geometry 1**	5	2/2 C+Ex	—
NMAG407	**Model Theory**	3	2/0 Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
	Optional and Elective Courses	26		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
	Optional and Elective Courses	36		

3 Degree Plans - Mathematics for Information Technologies

Coordinated by: Department of Algebra
Study branch coordinator: prof. RNDr. Aleš Drápal, CSc., DSc.

Specialization of the programme Mathematics for Information Technologies

This programme allows the student to specialize in two directions.

1. **Mathematics for information security.** This direction is focused on deepening the theoretical knowledge of number theory, probability theory, theory of error-correcting codes, complexity theory, theory of elliptic curves, and computer algebra applied to some of these subjects. Attention is also given to practical aspects such as internet security, standards in cryptography, and legal aspects of data security.
2. **Computer geometry.** This direction deepens theoretical knowledge in various algebraic and geometric subjects together with their applications in geometry of computer vision and robotics, computer graphics and image processing, optimization methods and numerical linear algebra.

Choice of a specialization

The direction is selected in three subsequent steps:

- *choice of the topic of Master thesis* at the beginning of Year 1
- *choice of elective courses*
- *choice of elective topics for the state final exam*

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Linear algebra, real analysis, and probability theory.
- Foundations of commutative and computer algebra (Galois theory, integral extensions, discrete Fourier transformation), modular arithmetic, multiplicative groups, finite fields, basic classes of error-correcting codes, and the group operations on elliptic curves.
- Basics of theoretic cryptography and geometric modelling. Programming in C.

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

3.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMMB403	**Computer Algebra 2**	6	3/1 C+Ex	—
NMMB405	**Complexity for Cryptography**	6	4/0 Ex	—
NMMB407	**Probability and Cryptography**	6	4/0 Ex	—
NMMB409	**Convex optimization**	9	4/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

3.2 Elective Courses

The elective courses for the specialization *Mathematics for Information Security* are denoted by (IS). The courses for specialization *Computer Geometry* are denoted by (CG). It is required to earn at least 45 credits from this group.

Code	Subject	Credits	Winter	Summer
NMMB401	Automata and Convolutional Codes (IS)	6	3/1 C+Ex	—
NMMB402	Numerical Algorithms (IS)	6	—	3/1 C+Ex
NMMB404	Cryptanalytic Attacks (IS)	6	—	3/1 C+Ex
NMMB501	Network Certification Security (IS)	5	2/2 C+Ex	—
NMAG436	Curves and Function Fields (IS)	6	—	4/0 Ex
NMMB431	Authentication Schemes (IS)*	3	—	2/0 Ex
NMMB436	Steganography and Digital Media (IS)	3	2/0 Ex	—
NMMB437	Legal Aspects of Data Protection (IS)	3	2/0 Ex	—
NMMB531	Number Field Sieve (IS)	3	2/0 Ex	—
NMMB532	Standards and Cryptography (IS)	3	—	2/0 Ex
NMMB533	Mathematical Software (IS) *	3	1/1 C+Ex	—
NMMB534	Quantum Information (IS)	6	—	3/1 C+Ex

NMMB538	Elliptic Curves and Cryptography (IS)	6	—	3/1 C+Ex
NMAG401	Algebraic Geometry (CG)	5	2/2 C+Ex	—
NMMB440	Geometry of Computer Vision (CG)	6	—	2/2 C+Ex
NMMB442	Geometric Problems in Robotics (CG)	6	—	2/2 C+Ex
NMAG563	Introduction to complexity of CSP (CG)	3	2/0 Ex	—
NMMB536	Optimization and Approximation CSP (CG)	6	—	2/2 C+Ex
NMNV531	Inverse Problems and Regularization (CG)	5	2/2 C+Ex	—
NMNV407	Matrix Iterative Methods 1 (CG)	6	4/0 Ex	—
NMNV438	Matrix Iterative Methods 2 (CG)	5	—	2/2 C+Ex
NMNV534	Numerical Optimization Methods (CG)	5	—	2/2 C+Ex
NMMB535	Compressed Sensing (CG)	6	2/2 C+Ex	—
NPGR013	Special Functions and Transformations in Image Processing (CG)	3	—	2/0 Ex
NPGR010	Computer Graphics III (CG)	6	2/2 C+Ex	—
NMMB433	Geometry for Computer Graphics (CG)	3	—	2/0 Ex
NPGR029	Variational methods in image processing (CG)	3	—	2/0 Ex
NMMB331	Boolean function (IS)	3	2/0 Ex	—
NMMB333	Introduction to data analysis (CG, IS)	5	2/2 C+Ex	—
NTIN104	Foundations of theoretical cryptography (IS)	5	—	2/1 C+Ex

*) These courses will not be taught any more.

3.3 State Final Exam

Requirements for taking the final exam

– Earning at least 120 credits during the course of the study.
– Completion of all obligatory courses prescribed by the study plan.
– Earning at least 45 credits by completion of elective courses.
– Submission of a completed Master's Thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of two subject areas. One question is asked from common subject area 1. Student chooses either two topic from among 2A, 2B, 2C in case of the specialization *Mathematics for information security*, or two topics from among 2D, 2E, 2F, 2G in case of specialization *Computer geometry.* One question is asked from every chosen topic.

1. Basic mathematical subjects.

Complexity classes and computational models, randomness and pseudorandomness, algorithms for algebraic structures, convex optimization.

2A. Information and Coding.

Classical and quantum information and its transfer. Consequences of quantum Fourier transform to cryptography. Convolution codes. Hidden a damaged information.

2B. Number theoretic algorithms.

Factorization: Pollard rho and Pollard p-1, CFRAC algorithm (including root approximation by chain fractions and solution of Pell equation), quadratic sieve (including Tonelli-Shanks algorithm). Basic methods for discrete logarithm: Pohlig-Hellman, Baby steps-Giant steps and index calculus.

2C. Elliptical curves.

basic properties of algebraic function fields and their groups of divisors. Weierstrass normal form of elliptic curve - equivalence and derivation. Picard group and addition of points on elliptic curve. Morphisms, endomorphisms and isogeny. Applications in cryptography.

2D. Computer vision and robotics.

Mathematical model of perspective camera. Calculation of movement of calibrated camera from the pictures of unknown scene. 3D reconstruction from two images of unknown scene. Geometry of three calibrated cameras. Denavit-Hartenberg description of kinematics of manipulator. Inverse kinematic problem of 6-arm serial manipulator - formulation and solution. Calibration of parameters of manipulator - formulation and solution.

2E. Image processing and computer graphics.

Modelling of inverse problems, regularization methods, digitization of image, deblurring, edge detection, image registration, compression, image synthesis, compressed sensing, analytical, kinematic and differential geometry.

2F. Approximation and optimization.

Convex optimization problems, duality, Lagrange dual problem. Algorithms for convex optimization, interior point method. Constraint satisfaction problem, algebraic approach to dichotomy conjecture. Weighted constraint satisfaction problem. Examples of computational problems formulated on wCSP, algebraic theory. Solution of problems with extremely large input.

2G. Numerical linear algebra.

LU and Choleski decomposition of a matrix, least squares, Krylov subspaces, matrix iterative methods (Arnoldi, Lanczos, joint gradients, generalized method of minimal residuum), QR algorithm, regularized methods for inverse problems, numerical stability.

3.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMMB405	**Complexity for Cryptography**	6	4/0 Ex	—
NMMB409	**Convex optimization**	9	4/2 C+Ex	—

NMMB403	**Computer Algebra 2**	6	3/1 C+Ex	—
NMMB407	**Probability and Cryptography**	6	4/0 Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
	Optional and Elective Courses	27		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
	Optional and Elective Courses	36		

4 Degree Plans - Mathematical Analysis

Coordinated by: Department of Mathematical Analysis
Study branch coordinator: doc. RNDr. Ondřej Kalenda, Ph.D., DSc.

The mathematical analysis curriculum offers advanced knowledge of fields traditionally forming mathematical analysis (real function theory, complex analysis, functional analysis, ordinary and partial differential equations). It is characterized by a depth of insight into individual topics and emphasis on their mutual relations and interconnections. Advanced knowledge of these topics is provided by a set of obligatory courses. Elective courses deepen the knowledge of selected fields, especially those related to the diploma thesis topic. Seminars provide contact with contemporary mathematical research. Mathematical analysis has close relationships with other mathematical disciplines, such as probability theory, numerical analysis and mathematical modelling. Students become familiar with these relationships in some of the elective courses. The programme prepares students for doctoral studies in mathematical analysis and related subjects. Applications of mathematical theory, theorems and methods to applied problems broaden the qualification to etemployment in a non-research environment.

The graduate will acquire advanced knowledge in principal fields of mathematical analysis (real function theory, complex analysis, functional analysis, ordinary and partial differential equations), understand their interconnections and relations to other mathematical disciplines. He/she will be able to apply advanced theoretical methods to real problems. The programme prepares students for doctoral studies but the knowledge and abilities acquired can be put into use in practical occupations as well.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Differential calculus of one and several real variables. Integral calculus of one real variable. Measure theory, Lebesgue measure and Lebesgue integral. Basic algebra (matrix calculus, vector spaces).
- Foundations of general topology (metric and topological spaces, completeness and compactness), complex analysis (Cauchy integral theorem, residue theorem, conformal maps) and functional analysis (Banach and Hilbert spaces, dual spaces, bounded operators, compact operators, basic theory of distributions).

- Elements of the theory of ordinary differential equations (basic properties of solutions and maximal solutions, linear systems, stability theory) and of partial differential equations (quasilinear first order equations, Laplace theorem and heat equation – fundamental solution and maximum principle, wave equation – fundamental solution, finite speed of wave propagation).

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

4.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMMA401	**Functional Analysis 1**	8	4/2 C+Ex	—
NMMA402	**Functional Analysis 2**	6	—	3/1 C+Ex
NMMA403	**Theory of Real Functions 1**	4	2/0 Ex	—
NMMA404	**Theory of Real Functions 2**	4	—	2/0 Ex
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NMMA407	**Ordinary Differential Equations 2**	5	2/2 C+Ex	—
NMMA408	**Complex Analysis 2**	5	—	2/2 C+Ex
NMMA501	**Nonlinear Functional Analysis 1**	5	2/2 C+Ex	—
NMMA502	**Nonlinear Functional Analysis 2**	5	—	2/2 C+Ex
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

4.2 Elective Courses

Set 1

The courses in this group introduce various research areas in mathematical analysis, illustrate their applications, and cover other fields that are related to mathematical analysis. It is required to earn at least 12 credits from this group.

Code	Subject	Credits	Winter	Summer
NMAG409	Algebraic Topology 1	5	2/2 C+Ex	—
NMAG433	Riemann Surfaces	3	2/0 Ex	—
NMMA433	Descriptive Set Theory 1	4	2/0 Ex	—
NMMA434	Descriptive Set Theory 2	4	—	2/0 Ex
NMMA435	Topological Methods in Functional Analysis 1	4	2/0 Ex	—
NMMA436	Topological Methods in Functional Analysis 2	4	—	2/0 Ex
NMMA437	Advanced Differentiation and Integration 1	4	2/0 Ex	—

NMMA438	Advanced Differentiation and Integration 2	4	—	2/0 Ex
NMMA440	Differential Equations in Banach Spaces	4	—	2/0 Ex
NMMA531	Partial Differential Equations 3	4	2/0 Ex	—
NMMA533	Introduction to Interpolation Theory 1	4	2/0 Ex	—
NMMA534	Introduction to Interpolation Theory 2	4	—	2/0 Ex
NMMO401	Continuum Mechanics	6	2/2 C+Ex	—
NMMO532	Mathematical Theory of Navier-Stokes Equations	3	—	2/0 Ex
NMMO536	Mathematical Methods in Mechanics of Compressible Fluids	3	—	2/0 Ex
NMNV405	Finite Element Method 1	5	2/2 C+Ex	—

Set 2

This group includes scientific seminars and workshops. It is required to earn at least 12 credits from this group. Each seminar yields 3 credits per semester and they can be taken repeatedly.

Code	Subject	Credits	Winter	Summer
NMMA431	Seminar on Differential Equations	3	0/2 C	0/2 C
NMMA451	Seminar on Geometrical Analysis	3	0/2 C	0/2 C
NMMA452	Seminar on Partial Differential Equations	3	0/2 C	0/2 C
NMMA454	Seminar on Function Spaces	3	0/2 C	0/2 C
NMMA455	Seminar on Real and Abstract Analysis	3	0/2 C	0/2 C
NMMA456	Seminar on Real Functions Theory	3	0/2 C	0/2 C
NMMA457	Seminar on Basic Properties of Function Spaces	3	0/2 C	0/2 C
NMMA458	Seminar on Topology	3	0/2 C	0/2 C
NMMA459	Seminar on Fundamentals of Functional Analysis	3	0/2 C	0/2 C

4.3 State Final Exam

Requirements for taking the final exam

– Earning at least 120 credits during the course of the study.
– Completion of all obligatory courses prescribed by the study plan.
– Earning at least 12 credits by completion of elective courses from group I.
– Earning at least 12 credits by completion of elective courses from group II.
– Submission of a completed master’s thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of five subject areas: “Real Analysis”, “Complex Analysis”, “Functional Analysis”, “Ordinary Differential Equations”, and “Partial Differential Equations”. One question is asked from each subject area.

Requirements for the oral part of the final exam

1. Real Analysis

Measure theory and signed measures, Radon measures. Absolutely continuous functions and functions with bounded variation. Hausdorff measure and Hausdorff dimension. Elements of descriptive set theory.

2. Complex Analysis

Meromorphic functions. Conformal mappings. Harmonic functions of two real variables. Zeros of holomorphic functions. Holomorphic functions of several complex variables. Analytic continuation.

3. Functional Analysis

Topological linear spaces. Locally convex spaces and weak topologies. Spectral theory in Banach algebras. Spectral theory of bounded and unbounded operators. Differential calculus in Banach spaces. Fixed points. Integral transformations. Theory of distributions.

4. Ordinary Differential Equations

Carathéodory theory of solutions. Systems of first order linear equations. Stability and asymptotical stability. Dynamical systems. Bifurcations.

5. Partial Differential Equations

Linear and quasilinear first order equations. Linear and nonlinear eliptic equations. Linear and nonlinear parabolic equations. Linear hyperbolic equations. Sobolev and Bochner spaces.

4.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMMA401	**Functional Analysis 1**	8	4/2 C+Ex	—
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMMA407	**Ordinary Differential Equations 2**	5	2/2 C+Ex	—
NMMA403	**Theory of Real Functions 1**	4	2/0 Ex	—
NMMA402	**Functional Analysis 2**	6	—	3/1 C+Ex
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NMMA408	**Complex Analysis 2**	5	—	2/2 C+Ex
NMMA404	**Theory of Real Functions 2**	4	—	2/0 Ex
	Optional and Elective Courses	10		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—

NMMA501	**Nonlinear Functional Analysis 1**	5	2/2 C+Ex	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
NMMA502	**Nonlinear Functional Analysis 2**	5	—	2/2 C+Ex
	Optional and Elective Courses	26		

5 Degree Plans - Numerical and Computational Mathematics

Coordinated by: Department of Numerical Mathematics
Study branch coordinator: doc. Mgr. Petr Knobloch, Dr.

This programme focuses on design, analysis, algorithmization, and implementation of methods for computer processing of mathematical models. It represents a transition from theoretical mathematics to practically useful results. An emphasis is placed on the creative use of information technology and production of programming applications. An integral part of the programme is the verification of employed methods. The students will study modern methods for solving partial differential equations, the finite element method, linear and non-linear functional analysis, and methods for matrix calculation. They will choose the elective courses according to the topic of their master's thesis. They can specialise in industrial mathematics, numerical analysis, or matrix computations.

The graduate will have attained the knowledge needed for numerical solution of practical problems from discretization through numerical analysis up to implementation and verification. He/she will be able to choose an appropriate numerical method for a given problem, conduct its numerical analysis, and implement its computation including analysis of numerical error. The graduate will be able to critically examine, assess, and tune the whole process of the numerical solution, and can assess the agreement between the numerical results and reality. He/she will be able to carry out an analytical approach to the solution of a general problem based on thorough and rigorous reasoning. The graduate will be qualified for doctoral studies and for employment in industry, basic or applied research, or government institutions.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Differential calculus for functions of one and several real variables. Integral calculus for functions of one variable. Measure theory, Lebesgue measure and Lebesgue integral. Basics of algebra (matrix calculus, vector spaces).
- Foundations of functional analysis (Banach and Hilbert spaces, duals, bounded operators, compact operators, basics of the theory of distributions), theory of ordinary differential equations (basic properties of the solution and maximal solutions, systems of linear equations, stability) and partial differential equations (quasilinear equations of first order, Laplace equation, heat equation and wave equation).
- Foundations of numerical mathematics (numerical quadrature, basics of the numerical solution of ordinary differential equations, finite difference method for partial differential equations) and of analysis of matrix computations (Schur theorem, orthogonal transformations, matrix decompositions, basic iterative methods).

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

5.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NMNV401	**Functional Analysis**	5	2/2 C+Ex	—
NMNV402	**Nonlinear Functional Analysis**	5	—	2/2 C+Ex
NMNV403	**Numerical Software 1**	5	2/2 C+Ex	—
NMNV404	**Numerical Software 2**	5	—	2/2 C+Ex
NMNV405	**Finite Element Method 1**	5	2/2 C+Ex	—
NMNV407	**Matrix Iterative Methods 1**	6	4/0 Ex	—
NMNV501	**Solution of Nonlinear Algebraic Equations**	5	2/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

5.2 Elective Courses

It is required to earn at least 28 credits from elective courses. The selection of elective courses should take into account the planned choice of the third subject area for the final exam. The subject area for which the course is recommended is shown in parentheses (3A, 3B or 3C). The course NMNV451 Seminar in Numerical Mathematics can be taken repeatedly. We recommend enrolling in it for each semester of study.

Code	Subject	Credits	Winter	Summer
NMNV436	Finite Element Method 2 (3B)	5	—	2/2 C+Ex
NMNV438	Matrix Iterative Methods 2 (3C)	5	—	2/2 C+Ex
NMNV451	Seminar in Numerical Mathematics	2	0/2 C	0/2 C
NMNV531	Inverse Problems and Regularization	5	2/2 C+Ex	—
NMNV532	Parallel Matrix Computations (3C)	5	—	2/2 C+Ex
NMNV533	Sparse Matrices in Direct Methods (3C)	5	2/2 C+Ex	—
NMNV534	Numerical Optimization Methods	5	—	2/2 C+Ex
NMNV535	Nonlinear Differential Equations (3B)	3	2/0 Ex	—
NMNV536	Numerical Solution of Evolutionary Equations (3A)	3	—	2/0 Ex
NMNV537	Mathematical Methods in Fluid Mechanics 1 (3A)	3	2/0 Ex	—
NMNV538	Mathematical Methods in Fluid Mechanics 2 (3A)	3	—	2/0 Ex

NMNV539	Numerical Solution of ODE (3B)	5	2/2 C+Ex	—
NMNV540	Fundamentals of Discontinuous Galerkin Method (3B)	3	—	2/0 Ex
NMNV541	Shape and Material Optimisation 1 (3A)	3	2/0 Ex	—
NMNV542	Shape and Material Optimisation 2 (3A)	3	—	2/0 Ex
NMNV543	Approximation Theory	4	2/1 C+Ex	—

5.3 State Final Exam

Requirements for taking the final exam

– Earning at least 120 credits during the course of the study.
– Completion of all obligatory courses prescribed by the study plan.
– Earning at least 28 credits by completion of elective courses.
– Submission of a completed master's thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of two common subject areas "1. Mathematical and Functional Analysis" and "2. Numerical Methods" and a choice of one of three subject areas "3A. Industrial Mathematics", "3B. Numerical Analysis", or "3C. Matrix Computations". One question is asked from each of the subject areas 1 and 2 and one question is asked from the subject area selected among 3A, 3B, or 3C.

Requirements for the oral part of the final exam

1. Mathematical and functional analysis

Partial differential equations, spectral analysis of linear operators, monotone and potential operators, solution of variational problems

2. Numerical methods

Finite element method, basic matrix iterative methods, methods for the solution of systems of nonlinear algebraic equations, basics of the implementation of numerical methods

3. Choice of one of the following topics:

3A. Industrial Mathematics

Mathematical methods in fluid mechanics, methods of material optimization, methods of solution of evolutionary equations

3B. Numerical Analysis

Nonlinear differential equations, numerical methods for ordinary differential equations, numerical solution of convection-diffusion problems

3C. Matrix Computations

Methods of Krylov subspaces, projections and problem of moments, connection between spectral information and convergence, direct methods for sparse matrices

5.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMNV407	**Matrix Iterative Methods 1**	6	4/0 Ex	—
NMNV401	**Functional Analysis**	5	2/2 C+Ex	—
NMNV403	**Numerical Software 1**	5	2/2 C+Ex	—
NMNV405	**Finite Element Method 1**	5	2/2 C+Ex	—
NMNV451	Seminar in Numerical Mathematics	2	0/2 C	—
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NMNV402	**Nonlinear Functional Analysis**	5	—	2/2 C+Ex
NMNV404	**Numerical Software 2**	5	—	2/2 C+Ex
NMNV451	Seminar in Numerical Mathematics	2	—	0/2 C
	Optional and Elective Courses	7		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NMNV501	**Solution of Nonlinear Algebraic Equations**	5	2/2 C+Ex	—
NMNV451	Seminar in Numerical Mathematics	2	0/2 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
NMNV451	Seminar in Numerical Mathematics	2	—	0/2 C
	Optional and Elective Courses	27		

6 Degree Plans - Mathematical Modelling in Physics and Technology

Coordinated by: Mathematical Institute of Charles University
Study branch coordinator: prof. RNDr. Josef Málek, CSc., DSc.

Mathematical modelling is an interdisciplinary field connecting mathematical analysis, numerical mathematics, and physics. The curriculum is designed to provide excellent basic knowledge in all these disciplines and to allow a flexible widening of knowledge by studying specialized literature when the need arises. All students take obligatory courses in continuum mechanics, partial differential equations, and numerical mathematics. Students will acquire the ability to design mathematical models of natural phenomena (especially related to continuum mechanics and thermodynamics), analyse them, and conduct numerical simulations. After passing the obligatory classes, students get more closely involved with physical aspects of mathematical modelling (model design), with mathematical analysis of partial differential equations, or with methods for computing mathematical models. The grasp of all levels of mathematical modelling (model, analysis, simulations) allows the students to use modern results from all relevant fields to address problems in physics, technology, biology, and medicine that surpass the scope of the fields individually. Graduates will be able to pursue academic or commercial careers in applied mathematics, physics and technology.

The graduate will have mastered methods and results in continuum mechanics and thermodynamics, mathematical analysis of partial differential equations, and numerical mathematics, and will be ready to widen his/her knowledge by studying specialized literature. He/she will be able to formulate questions regarding the physical substance of natural phenomena, especially those related to the behaviour of fluids and solid matter in the framework of classical physics, with applications to technology, medicine, biology, geophysics, and meteorology. He/she will be able to choose appropriate mathematical models for such phenomena, carry out its mathematical analysis, and conduct numerical simulations with suitable methods. He/she will be able to critically analyse, evaluate, and tie in the whole modelling process. In simpler cases, he/she will be able to assess the errors in the modelling process and predict the agreement between numerical results and the physical process. The graduate will be ready to work in interdisciplinary teams. He/she will be able to pose interesting questions in a format ready for mathematical investigation and use abstract mathematical results in order to address applied problems.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Foundations of differential and integral calculus of one variable and several variables. Volume, surface and line integral. Measure theory. Lebesgue integral.
- Foundations of linear algebra (vector spaces, matrices, determinants, Jordan canonical form, eigenvalues and eigenvectors, multilinear algebra, quadratic forms). Numerical solution of systems of linear algebraic equations (Schur theorem, QR decomposition, LU decomposition, singular value decomposition, least squares problem, partial eigenvalue problem, conjugate gradient method, GMRES, backward error, sensitivity and numerical stability, QR algorithm).
- Foundations of complex analysis (Cauchy theorem, residual theorem, conformal mappings, Laplace transform).
- Foundations of functional analysis and theory of metric spaces (Banach and Hilbert spaces, operators and functionals, Hahn-Banach theorem, dual space, bounded operators, compact operators, theory of distributions).
- Foundations of theory of ordinary differential equations (existence of solution, maximal solution, systems of linear equations, stability) and partial differential equations (quasilinear first order equations, Laplace equation and heat equation – fundamental solution and maximum principle, wave equation – fundamental solution, finite propagation speed).
- Foundations of classical mechanics (Newton laws, Lagrange equations, Hamilton equations, variational formulation, rigid body dynamics).

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

6.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMMA401	**Functional Analysis 1**	8	4/2 C+Ex	—
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NMMO401	**Continuum Mechanics**	6	2/2 C+Ex	—
NMMO402	**Thermodynamics and Mechanics of Non-Newtonian Fluids**	5	—	2/1 C+Ex
NMMO403	**Computer Solutions of Continuum Physics Problems**	5	—	2/2 C+Ex
NMMO404	**Themodynamics and Mechanics of Solids**	5	—	2/1 C+Ex
NMNV405	**Finite Element Method 1**	5	2/2 C+Ex	—
NMNV407	**Matrix Iterative Methods 1**	6	4/0 Ex	—
NOFY036	**Thermodynamics and Statistical Physics**	6	3/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

6.2 Elective Courses

It is required to earn at least 16 credits from elective courses.

Code	Subject	Credits	Winter	Summer
NMMA407	Ordinary Differential Equations 2	5	2/2 C+Ex	—
NMMA531	Partial Differential Equations 3	4	2/0 Ex	—
NMMO432	Classical Problems of Continuum Mechanics	4	—	2/1 C+Ex
NMMO531	Biothermodynamics	5	2/2 C+Ex	—
NMMO532	Mathematical Theory of Navier-Stokes Equations	3	—	2/0 Ex
NMMO533	Nonlinear Differential Equations and Inequalities 1	6	3/1 C+Ex	—
NMMO534	Nonlinear Differential Equations and Inequalities 2	6	—	3/1 C+Ex
NMMO535	Mathematical Methods in Mechanics of Solids	3	2/0 Ex	—
NMMO536	Mathematical Methods in Mechanics of Compressible Fluids	3	—	2/0 Ex
NMMO537	Saddle Point Problems and Their Solution	5	—	2/2 C+Ex
NMMO539	Mathematical Methods in Mechanics of Non-Newtonian Fluids	3	2/0 Ex	—
NMMO541	Theory of Mixtures	4	2/1 C+Ex	—

NMNV403	Numerical Software 1	5	2/2 C+Ex	—
NMNV404	Numerical Software 2	5	—	2/2 C+Ex
NMNV501	Solution of Nonlinear Algebraic Equations	5	2/2 C+Ex	—
NMNV532	Parallel Matrix Computations	5	—	2/2 C+Ex
NMNV537	Mathematical Methods in Fluid Mechanics 1	3	2/0 Ex	—
NMNV538	Mathematical Methods in Fluid Mechanics 2	3	—	2/0 Ex
NOFY026	Classical Electrodynamics	6	—	2/2 C+Ex
NTMF034	Electromagnetic Field and Special Theory of Relativity	5	—	2/1 Ex

6.3 State Final Exam

Requirements for taking the final exam

- Earning at least 120 credits during the course of the study.
- Completion of all obligatory courses prescribed by the study plan.
- Earning at least 16 credits by completion of elective courses.
- Submission of a completed master's thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of three subject areas: "Continuum Mechanics and Thermodynamics", "Functional Analysis and Partial Differential Equations", and "Numerical Methods". One question is asked from each subject area.

Requirements for the oral part of the final exam

1. Continuum mechanics and thermodynamics

Kinematics. Stress tensor. Balance equations. Constitutive relations. Models for fluids and solids.

2. Functional analysis and partial differential equations

Linear operators and functionals, compact operators. Distributions. Function spaces. Weak solutions of the linear elliptic, parabolic and hyperbolic second order partial differential equations – foundations of the existence theory, elementary theory of qualitative properties of the solutions.

3. Numerical methods

Numerical methods for partial differential equations. Finite element method. Iterative methods for solving systems of linear algebraic equations.

6.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMMA401	**Functional Analysis 1**	8	4/2 C+Ex	—
NMMA405	**Partial Differential Equations 1**	6	3/1 C+Ex	—
NMMO401	**Continuum Mechanics**	6	2/2 C+Ex	—

NOFY036	**Thermodynamics and Statistical Physics**	6	3/2 C+Ex	—
NMNV405	**Finite Element Method 1**	5	2/2 C+Ex	—
NMMA406	**Partial Differential Equations 2**	6	—	3/1 C+Ex
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NMMO402	**Thermodynamics and Mechanics of Non-Newtonian Fluids**	5	—	2/1 C+Ex
NMMO403	**Computer Solutions of Continuum Physics Problems**	5	—	2/2 C+Ex
NMMO404	**Themodynamics and Mechanics of Solids**	5	—	2/1 C+Ex
	Optional and Elective Courses	1		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NMNV407	**Matrix Iterative Methods 1**	6	4/0 Ex	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
	Optional and Elective Courses	30		

7 Degree Plans - Probability, Mathematical Statistics and Econometrics

Coordinated by: Department of Probability and Mathematical Statistics
Study branch coordinator: doc. RNDr. Daniel Hlubinka, Ph.D.

The curriculum is targeted at students who wish to obtain theoretical and practical knowledge about the mathematics of random events. It is primarily characterized by a balance between rigorous mathematical theory, depth of insight into various fields of the subject (probability, statistics, econometrics), and applications in various areas. The students will obtain a general background by taking compulsory courses in probability, optimization, linear regression and random processes. They continue by taking elective courses to extend their expertise and choose a specialization they wish to study more deeply. In seminars, they learn to work independently as well as to collaborate on complex projects. Great emphasis is placed on the development of independent analytical thinking. Probability, statistics and econometrics have a close relationship to other mathematical subjects (mathematical analysis, numerical mathematics, discrete mathematics). Applications are inspired by real problems from economics, medicine, technology, natural sciences, physics and computer science. The primary objective of the programme is to prepare students for successful careers in academia as well as in finance, telecommunications, marketing, medicine and natural sciences.

The graduate will be familiar with mathematical modelling of random events and processes and its applications. He/she will understand the foundations of probability theory, mathematical statistics, random process theory and optimization. His/her general background will have been extended to a deeper knowledge of random process theory and stochastic analysis, modern statistical methods, or advanced optimization and time series analysis. The graduate will understand the substance of the methods, grasp their mutual relationships, and will be able to actively extend them and use them. He/she will know how to apply theoretical knowledge to practical problems in a creative way. The graduate's ability to think logically, to analyse problems, and to solve non-trivial problems can be put to use in independent and creative jobs in the commercial sector or in academic positions.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Differential and integral calculus of one variable and several variables. Measure theory. Lebesgue integral. Vector spaces, matrix algebra. Foundations of complex and functional analysis.
- Foundations of probability theory.
- Foundations of mathematical statistics and data analysis.
- Markov chain theory.

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

7.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMSA401	**Primary Seminar**	2	0/2 C	—
NMSA403	**Optimisation Theory**	5	2/2 C+Ex	—
NMSA405	**Probability Theory 2**	5	2/2 C+Ex	—
NMSA407	**Linear Regression**	8	4/2 C+Ex	—
NMSA409	**Stochastic Processes 2**	8	4/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

7.2 Elective Courses

Set 1

It is required to earn at least 7 credits from this group.

Code	Subject	Credits	Winter	Summer
NMEK450	Econometrics Seminar 1	2	—	0/2 C
NMEK551	Econometric Project Seminar	5	0/2 C	—
NMST450	Statistical Seminar 1	2	—	0/2 C
NMST551	Statistical Workshop	5	0/2 C	—
NMTP450	Seminar on Probability 1	2	—	0/2 C
NMTP551	Seminar on Probability 2	5	0/2 C	—

Set 2

It is required to earn at least 43 credits from this group. We recommend making a planned choice of subject areas for the final exam and the master's thesis topic when choosing elective courses.

Code	Subject	Credits	Winter	Summer
NMEK432	Econometrics	8	—	4/2 C+Ex
NMEK436	Computational Aspects of Optimisation	5	—	2/2 C+Ex
NMEK531	Mathematical Economics	5	2/2 C+Ex	—
NMEK532	Optimisation with Applications to Finance	8	—	4/2 C+Ex
NMFM431	Investment Analysis	5	2/2 C+Ex	—
NMFM437	Mathematics in Finance and Insurance	6	4/0 Ex	—
NMFM531	Financial Derivatives 1	3	2/0 Ex	—
NMFM532	Financial Derivatives 2	3	—	2/0 Ex
NMFM535	Stochastic Analysis in Financial Mathematics	5	—	2/2 C+Ex
NMFM537	Credit Risk in Banking	3	2/0 Ex	—
NMST431	Bayesian Methods	5	2/2 C+Ex	—
NMST432	Advanced Regression Models	8	—	4/2 C+Ex
NMST434	Modern Statistical Methods	8	—	4/2 C+Ex
NMST436	Experimental Design	5	2/2 C+Ex	—
NMST438	Survey Sampling	5	2/2 C+Ex	—
NMST440	Computational Environment for Statistical Data Analysis	4	—	0/2 C
NMST442	Matrix Computations in Statistics	5	—	2/2 C+Ex
NMST531	Censored Data Analysis	5	2/2 C+Ex	—
NMST532	Design and Analysis of Medical Studies	5	—	2/2 C+Ex
NMST533	Asymptotic Inference Methods	3	2/0 Ex	—
NMST535	Simulation Methods	5	—	2/2 C+Ex
NMST537	Time Series	8	4/2 C+Ex	—
NMST539	Multivariate Analysis	5	—	2/2 C+Ex
NMST541	Statistical Quality Control	5	—	2/2 C+Ex
NMST543	Spatial Statistics	5	2/2 C+Ex	—
NMST552	Statistical Consultations	2	—	0/2 C
NMTP432	Stochastic Analysis	8	—	4/2 C+Ex
NMTP434	Invariance Principles	6	—	4/0 Ex
NMTP436	Continuous Martingales and Counting Processes	3	—	2/0 Ex
NMTP438	Spatial Modelling	8	—	4/2 C+Ex
NMTP532	Ergodic Theory	4	—	3/0 Ex
NMTP533	Applied Stochastic Analysis	5	2/2 C+Ex	—
NMTP535	Selected Topics on Measure Theory	3	2/0 Ex	—

NMTP537	Limit Theorems for Sums of Random Variables	3	2/0 Ex	—
NMTP539	Markov Chain Monte Carlo Methods	5	2/2 C+Ex	—
NMTP541	Stochastic Geometry	3	—	2/0 Ex
NMTP543	Stochastic Differential Equations	6	4/0 Ex	—
NMTP545	Theory of Probability Distributions	3	2/0 Ex	—

7.3 State Final Exam

Requirements for taking the final exam

– Earning at least 120 credits during the course of the study.
– Completion of all obligatory courses prescribed by the study plan.
– Earning at least 7 credits by completion of elective courses from group I.
– Earning at least 43 credits by completion of elective courses from group II.
– Submission of a completed master's thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of three subject areas. The first subject area is common. The second subject area is selected from three options (2A, 2B, 2C). The third subject area is selected from seven options 3A–3G. One question is asked from the common subject area and one from each selected optional subject area.

Requirements for the oral part of the final exam

Common subject area

1. Foundations of Probability, Statistics and Random Processes

Foundations of Markov chain theory. Stationary sequences and processes. Linear regression model. Conditional expectation. Martingales in discrete time. Optimization, linear and non-linear programming.

Optional subject area 2: Advanced Models

A choice of one of three options

2A. Econometrics and Optimization Methods

Stationary sequences, time series. Foundations of econometrics. Advanced optimization.

2B. Advanced Statistical Analysis.

Modern theory of estimation and statistical inference. Regression models for non-normal and correlated data.

2C. Processes in Time and Space.

Stochastic processes in continuous time. Martingales. Invariance principles. Brownian motion.

Optional subject area 3: Special Topics

A choice of one of seven options

3A. Econometric Models

Mathematical economics. Time series with financial applications. Advanced econometrical and statistical models. Multivariate statistical analysis.

3B: Optimization Methods

General optimization problems, optimal control. Applications of optimization in economics and finance. Mathematical economics. Time series.

3C: Spatial Modelling

Spatial modelling and spatial statistics. Foundations of stochastic analysis. Limit theorems in probability theory.

3D: Stochastic Analysis

Stochastic analysis. Itô formula. Stochastic differential equations. Poisson processes, stationary point processes. Limit theorems.

3E. Statistics in Industry, Trade and Business

Survey sampling. Design of industrial experiments. Time series. Statistical quality control. Reliability theory.

3F. Statistics in Natural Sciences

Design and analysis of medical experiments. Multivariate statistical analysis. Survival analysis. Bayesian methods.

3G. Theoretical Statistics

Invariance principles. Limit theorems. Methods for censored data analysis. Multivariate analysis.

7.4. Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMSA407	**Linear Regression**	8	4/2 C+Ex	—
NMSA409	**Stochastic Processes 2**	8	4/2 C+Ex	—
NMSA403	**Optimisation Theory**	5	2/2 C+Ex	—
NMSA405	**Probability Theory 2**	5	2/2 C+Ex	—
NMSA401	**Primary Seminar**	2	0/2 C	—
	Optional and Elective Courses	32		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ023	**Diploma Thesis I**	6	0/4 C	—
NSZZ024	**Diploma Thesis II**	9	—	0/6 C
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
	Optional and Elective Courses	30		

8 Degree Plans - Financial and Insurance Mathematics

Coordinated by: Department of Probability and Mathematical Statistics
Study branch coordinator: prof. RNDr. Tomáš Cipra, DrSc.

This programme provides theoretical and applied education in financial and insurance mathematics. A solid mathematical background provides the foundations for developing disciplines of mathematical modelling in the insurance and banking industry and other financial areas. The graduate will be able to develop financial and insurance products and analyse their profitability and risk.

The graduate will have a deep knowledge of basic mathematical disciplines (mathematical analysis, algebra) and special knowledge in the fields of probability and statistics, stochastic processes, mathematical methods in finance, life and non-life insurance, advanced financial management, risk theory, accounting, and modelling with progressive systems (Mathematica). This knowledge provides tools for effective modelling of financial and insurance products, analysis of their profitability and risk, and other characteristics important for effective financial management.

Assumed knowledge

It is assumed that an incoming student of this branch has sufficient knowledge of the following topics and fields:

- Multivariate differential and integral calculus, measure theory, Lebesgue integral, vector spaces, matrix algebra.
- Foundations of probability theory, mathematical statistics, and data analysis. Theory of Markov chains.
- Foundations of financial mathematics and accounting.
- Foundations of functional programming.

Should an incoming student not meet these entry requirements, the coordinator of the study programme may assign a method of acquiring the necessary knowledge and abilities, which may for example mean taking selected bachelor's courses, taking a reading course with an instructor, or following tutored independent study.

8.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NMFM401	**Mathematics of Non-Life Insurance 1**	5	2/2 C+Ex	—
NMFM402	**Mathematics of Non-Life Insurance 2**	5	—	2/2 C+Ex
NMFM404	**Selected Software Tools for Finance and Insurance**	3	—	2/0 Ex
NMFM405	**Life Insurance 1**	5	2/2 C+Ex	—
NMFM406	**Life Insurance 2**	3	—	2/0 Ex
NMFM408	**Probability for Finance and Insurance**	3	—	2/0 Ex
NMFM410	**Insurance Companies Accounting**	5	—	2/2 C+Ex
NMFM416	**Life Insurance 2, exercises**	2	—	0/2 C
NMFM501	**Seminar on Actuarial Sciences 1**	2	0/2 C	—
NMFM502	**Seminar on Actuarial Sciences 2**	1	—	0/2 C
NMFM503	**Risk Theory**	8	4/2 C+Ex	—
NMFM505	**Stochastic Models for Finance and Insurance**	5	—	2/2 C+Ex

NMFM507	**Advanced Topics of Financial Management**	2	2/0 Ex	—
NMSA407	**Linear Regression**	8	4/2 C+Ex	—
NMSA409	**Stochastic Processes 2**	8	4/2 C+Ex	—
NMST537	**Time Series**	8	4/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

8.2 Elective Courses

It is required to earn at least 5 credits from the following elective courses.

Code	Subject	Credits	Winter	Summer
NMFM431	Investment Analysis	5	2/2 C+Ex	—
NMFM531	Financial Derivatives 1	3	2/0 Ex	—
NMFM532	Financial Derivatives 2	3	—	2/0 Ex
NMSA403	Optimisation Theory	5	2/2 C+Ex	—
NMST531	Censored Data Analysis	5	2/2 C+Ex	—
NMST539	Multivariate Analysis	5	—	2/2 C+Ex

Recommended Optional Courses

Code	Subject	Credits	Winter	Summer
NMEK432	*Econometrics*	8	—	4/2 C+Ex
NMFM462	*Practical Aspects of Financial Risk Measuring and Management*	3	2/0 Ex	—
NMFM535	*Stochastic Analysis in Financial Mathematics*	5	—	2/2 C+Ex

8.3 State Final Exam

Requirements for taking the final exam

– Earning at least 120 credits during the course of the study.
– Completion of all obligatory courses prescribed by the study plan.
– Earning at least 5 credits by completion of elective courses.
– Submission of a completed master's thesis by the submission deadline.

Oral part of the state final exam

The oral part of the final exam consists of three subject areas: "Applied Probability", "Insurance", and "Finance and Accounting". One question is asked from each subject area.

Requirements for the oral part of the final exam

1. Probability and Statistics

Random variables, characteristics of their distributions. Random vectors, joint distribution, covariance, correlation, dependence modelling and measurement. Conditional distribution. Probability distributions in financial mathematics. Parameter estimators and their properties. Confidence intervals. Hypothesis testing principles. Maximum likelihood principle and the method of moments. One-sample, two-sample, paired tests. Analysis of variance. Linear regression model. The Bayes principle. Laws of large numbers and central limit theorems. Markov chains. Stationary processes. Time Series. Credibility theory. Collective risk models. Essentials of stochastic analysis.

2. Insurance

Demographic model of life insurance. Capital and life policies. Reserves of life insurance. Multiple decrement models. Multiple life insurance. Solvency and reinsurance. Technical reserves of general insurance. Tariff rates.

3. Finance and Accounting

Foundations of finance. Securities and their pricing. Financial risk. Stock market analysis. Accounting.

8.4 Recommended Course of Study

1st year

Code	Subject	Credits	Winter	Summer
NMSA407	**Linear Regression**	8	4/2 C+Ex	—
NMSA409	**Stochastic Processes 2**	8	4/2 C+Ex	—
NMFM401	**Mathematics of Non-Life Insurance 1**	5	2/2 C+Ex	—
NMFM405	**Life Insurance 1**	5	2/2 C+Ex	—
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NMFM402	**Mathematics of Non-Life Insurance 2**	5	—	2/2 C+Ex
NMFM410	**Insurance Companies Accounting**	5	—	2/2 C+Ex
NMFM404	**Selected Software Tools for Finance and Insurance**	3	—	2/0 Ex
NMFM406	**Life Insurance 2**	3	—	2/0 Ex
NMFM408	**Probability for Finance and Insurance**	3	—	2/0 Ex
NMFM416	**Life Insurance 2, exercises**	2	—	0/2 C
	Optional and Elective Courses	7		

2nd year

Code	Subject	Credits	Winter	Summer
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NMFM503	**Risk Theory**	8	4/2 C+Ex	—
NMST537	**Time Series**	8	4/2 C+Ex	—
NMFM501	**Seminar on Actuarial Sciences 1**	2	0/2 C	—
NMFM505	**Stochastic Models for Finance and Insurance**	5	—	2/2 C+Ex

NMFM507	**Advanced Topics of Financial Management**	2	2/0 Ex	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C
NMFM502	**Seminar on Actuarial Sciences 2**	1	—	0/2 C
	Optional and Elective Courses	10		

Bachelor of Computer Science

1 General Information

Programme Coordinator: doc. RNDr. Ondřej Čepek, Ph.D.

Study specializations

The Bachelor of Computer Science programme has a common first year of study and is divided into six specializations starting in the second year of study:

– General Computer Science
– Programming and Software Development
– Systems Programming
– Databases and Web
– Artificial Intelligence
– Computer Graphics, Vision and Game Development.

Students select their specialization during the second year of their study in accordance with the study regulations. Please note that some specializations might not be offered; in 2020/2021 we currently plan to offer General Computer Science, Databases and Web, and Artificial Intelligence.

Degree plans

The course of study in the individual specializations is regulated by the relevant degree plan, which specifies the obligatory and elective courses, the requirements for the State Final Exam, and a recommended course of study. The elective courses are in each specialization divided into several groups. A minimum number of credits should be obtained from elective courses overall; in addition, a minimum total number of credits is also required for certain groups of elective courses. Besides obligatory courses and the required number of elective courses, each student may sign up for additional courses taught at our faculty or at other faculties of Charles University (these are called “optional courses”).

All six specializations share a large part in common, containing obligatory courses that cover the foundations of mathematics, theoretical computer science, programming, and software systems. Most of these subjects are recommended for the first year in the entire Computer Science programme. The recommended course of study for the first year specified below consists of obligatory courses (in boldface) and several optional courses (in italics). Of course, other optional courses may be selected instead of those that are recommended, provided that a total of at least 60 credits is achieved within the first academic year.

Recommended course of study for the first year

Code	Subject	Credits	Winter	Summer
NPRG062	**Introduction to Algorithms**	4	2/1 C+Ex	—

NPRG030	**Programming 1**	5	2/2 C	—
NSWI120	**Principles of Computers**	3	2/0 Ex	—
NSWI141	**Introduction to Networking**	3	2/0 MC	—
NDMI002	**Discrete Mathematics**	5	2/2 C+Ex	—
NMAI057	**Linear Algebra 1**	5	2/2 C+Ex	—
NMAI069	*Mathematical skills* [1]	2	0/2 C	—
NTVY014	*Physical Education I* [2]	1	0/2 C	—
ASE500129	*Czech Language Course 1* [3]	3	0/2 C	—
NTIN060	**Algorithms and Data Structures 1**	5	—	2/2 C+Ex
NPRG031	**Programming 2**	5	—	2/2 C+Ex
NSWI170	**Computer Systems**	5	—	2/2 C+Ex
NSWI177	**Introduction to Linux**	4	—	1/2 MC
NMAI054	**Mathematical Analysis 1**	5	—	2/2 C+Ex
NMAI058	**Linear Algebra 2**	5	—	2/2 C+Ex
NTVY015	*Physical Education II* [2]	1	—	0/2 C
ASE500130	*Czech Language Course 2* [3]	3	—	0/2 C

[1] The course NMAI069 Mathematical Skills is designed for students who wish to gain and practice the fundamental mathematical skills needed for the more mathematically oriented courses given at our faculty. Emphasis is put on the ability to use precise and correct mathematical formulations and on basic proof techniques.

[2] The Physical Education courses are obligatory for students on the programme taught in Czech, while they are elective for students on the programme taught in English. If you like sports, this may be a course for you, but there is no obligation to take it.

[3] The Czech Language Courses are optional, offered as a counterpart to the elective English Language Courses recommended for students studying in the programme taught in Czech. Since these courses are elective, they may naturally be replaced by any other course while maintaining the minimum of 30 credits per semester.

Some obligatory courses common to all specializations are taught in the second and third year of study. They are listed below.

Common obligatory courses in the second and third year of study

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1**	5	2/2 C+Ex	—
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project** [4]	4	—	0/1 C
NSZZ031	**Bachelor Thesis**	6	—	0/4 C

[4] It is possible to sign up for the course NPRG045 both in the winter semester and in the summer semester; the standard period is the summer semester.

Each individual specialization requires additional obligatory courses and groups of elective courses. A detailed degree plan for each specialization is given later in this text.

Recommended course of study for the second and third year

The recommended course of study is prepared for each specialization in such a way that the obligatory courses are scheduled in the required order, the student obtains in time the credits needed for enrolment in the next year of study, and the student fulfils in time all the prerequisites needed in order to take the State Final Exam. A recommended course of study for each specialization is given later in this text.

Branches within specializations

Some specializations are further divided into branches. Individual branches within the same specialization differ only in one area of prerequisites for the State Final Exam. Students should adjust their choice of elective and optional courses according to the branch in which they intend to take the State Final Exam. The choice of a particular branch within the student's specialization is declared only when signing up for the State Final Exam.

State Final Exam

The State Final Exam consists of two parts:

– Defence of Bachelor Thesis
– Exam in Mathematics and Computer Science

Each part of the State Final Exam is graded. The final grade for the State Final Exam is determined by the grades obtained for each part. The student can sign up for each part of the State Final Exam separately. Bachelor studies are successfully concluded only upon passing both parts of the State Final Exam. In case of failure, the student retakes those parts of the State Final Exam which he or she failed. Each part of the State Final Exam can be retaken at most twice.

Necessary conditions for signing up for either part of the State Final Exam are the following:

– passing all the obligatory courses of a given specialization,
– obtaining the required number of credits for elective courses,
– submitting a completed bachelor thesis by the specified deadline (necessary for signing up for the bachelor thesis defence),
– obtaining at least 180 credits (necessary for signing up for the last part of the State Final Exam).

A bachelor thesis topic is typically assigned at the beginning of the third year. The bachelor thesis usually consists of either a software package, which may be a continuation of the Individual Software Project (see degree plans above), or a piece of theoretical work. We recommend choosing a topic offered by the department which is connected with the selected specialization. In case another topic (offered by another department or suggested by the student) is to be selected, we strongly recommend consulting the relevant Specialization Coordinator before doing so.

The prerequisites for the State Final Exam are divided into two parts, one common to all specializations and the other specific to the given specialization. The list of common prerequisites is given below this paragraph; the prerequisites specific to the various specializations are listed after their degree plans given further below.

Knowledge requirements for the State Final Exam common to all specializations

Mathematics

1. Fundamentals of Differential and Integral Calculus

Sequences and series of numbers and their properties (properties of limits and sums, convergence criteria). Real functions of one variable. Continuity, limit of a function (ordinary, infinite). Derivatives: definition and basic rules. Applications (examination of behaviour of functions, Taylor polynomial with remainder). Antiderivatives (primitive functions): definition, uniqueness, existence, methods of calculation.

2. Algebra and Linear Algebra

Groups and subgroups, definitions, examples, commutativity. Fields – definition, characterization of a field, finite fields. Vector spaces and subspaces, properties, basic notions (linear combination, linear span/ hull, generators, linear dependence and independence, basis, dimension, coordinates) and uses. Practical ability in testing for linear dependence and independence, finding a basis, determining the dimension etc. Scalar product and its properties. Norm and its relation to scalar product, examples. Orthogonality, othonormal basis, properties and uses (e.g. finding coordinates, projection). Systems of linear equations and sets of solutions. Solution methods, Gauss and Gauss–Jordan elimination, reduced row echelon form of a matrix and uniqueness (without proof). Matrices, operations with matrices (addition, multiplication, transposition, etc.), interpretation of matrix multiplication as composition of linear mappings. Matrix rank, rank of a transposition. Eigenvalues and eigenvectors of a matrix, geometric meaning and properties, spectral radius. Characteristic polynomial, relationship between eigenvalues and roots of polynomials.

3. Discrete Mathematics

Relations, properties of binary relations (reflexivity, symmetry, antisymmetry, transitivity). Equivalence relations and partition into equivalence classes. Partial orders, basic concepts (minimal and maximal elements, minimum and maximum elements, chains, antichains), height and width of a partially ordered set and their mutual relationship. Functions, types of functions (injective, surjective, bijective) and the number of functions between two finite sets of a given type. Permutations and their basic properties (number of permutations, fixed points, etc.). Binomial coefficients and relationships between them, binomial theorem and its applications. Principle of inclusion and exclusion, general formulation (and proof), applications (Euler's totient function, number of surjections, etc.). Hall's theorem on systems of distinct representatives (SDR) and its relation to matchings in a bipartite graph, principle of proof and algorithmic consequences (polynomial-time algorithm for finding an SDR).

4. Graph Theory

Basic concepts (graph, vertices and edges, graph isomorphism, subgraph, vertex neighbourhood and vertex degree, graph complement, bipartite graph), basic examples of graphs (complete graph, complete bipartite graph, paths, cycles). Connected graphs, connected components, distance in a graph. Trees: definition and basic properties (existence of leaves, number of edges), equivalent characterizations of trees. Planar graphs: definition and basic concepts (planar graph and plane drawing of a graph, faces), Euler's formula and the maximum number of edges in a planar graph (proof and applications). Graph colourings: definition of a proper colouring, relation between chromatic number and clique number. Edge- and vertex-connectivity, edge and vertex version of Menger's theorem. Directed graphs: weak and strong connectivity. Network flows: definition of a network and of a flow, existence of a maximum flow (without proof), basic principle of finding a maximum flow in a network with integer-valued capacities (e.g. using the Ford–Fulkerson algorithm).

5. Probability and Statistics

Random events, conditional probability, independence of random events – definitions of these terms, Bayes' formula, applications. Random variables, mean (expectation), distribution of random variables, geometric, binomial and normal distribution. Linear combination of random variables – linearity of expectation, applications. Point estimates, confidence intervals, hypothesis testing.

6. Logic

Syntax – knowledge of and working with the basic syntax of propositional and predicate logic (language, open and closed formulas, etc.). Normal forms of propositional formulas, prenex normal forms of predicate logic formulas – knowledge of basic normal forms (CNF, DNF, PNF), converting to normal form, applications in algorithms (SAT, resolution). Semantics, truth, falsity, independence of a formula with respect to a theory, satisfiability, tautologies, logical consequence, the notion of a model of a theory, extensions of theories.

Computer Science

1. Automata and Languages

Regular languages: finite automaton, language accepted by a finite automaton, deterministic, nondeterministic, epsilon transitions, regular expressions, Kleene's theorem, iteration (pumping) lemma for finite automata, regular grammars. Context-free languages: context-free grammar, language generated by a grammar, push-down automaton, class of languages accepted by push-down automata. Turing machine: type 0 grammar, diagonal language, universal language. Chomsky hierarchy: determination of equivalence or inclusion between classes of languages generated by the automata and grammars mentioned above. Classifying a given language in the Chomsky hierarchy (typically by constructing an appropriate automaton or grammar and proving that the language does not belong to a lower class using an iteration lemma).

2. Algorithms and Data Structures

Time complexity of algorithms: time and space needed for computation on a given input, time and space complexity of an algorithm, data size measurement, complexity in the worst, best and average case, asymptotic notation. Complexity classes: P and NP, problem reducibility, NP-hardness and NP-completeness, examples of NP-complete problems and reductions among them. "Divide and conquer" technique: recursive division of a problem into subproblems, complexity computation using recursive equations, master theorem, applications (Merge sort, multiplication of long numbers, Strassen's algorithm). Binary search trees: definition of a search tree, operations with non-balanced trees, AVL trees (definition only). Heaps: binary heap. Hashing: hashing with buckets, open addressing. Sorting: primitive sorting algorithm (Bubble sort, Insertion sort etc.), Heapsort, Quicksort, lower bound for sorting based on pairwise comparisons, bucket sorting for numbers and strings. Graph algorithms: depth-first search and breadth-first search, connected component detection, topological sorting of directed graphs, shortest paths in weighted graphs (Dijkstra's algorithm, Bellman–Ford algorithm), minimum spanning trees (Jarník's algorithm, Borůvka's algorithm), network flows (Ford–Fulkerson algorithm). Algebraic algorithms: Euclid's algorithm.

3. Programming Languages

Concepts and principles of object-oriented design: classes, interfaces, methods, attributes, inheritance (visibility of members, namespaces, separation into packages/modules), multiple inheritance and its problems (language-specific methods for resolving problems, multiple and virtual inheritance in C++, single inheritance and default methods in Java), implementing interfaces, polymorphism (static vs. dynamic polymorphism), functional elements of object-oriented languages (function objects, lambdas, support in standard libraries). Implementation of object-oriented languages: basic object-oriented concepts in a concrete language (Java, C++, C#), primitive types vs. objects (implementation of primitive types, memory representation of compound types and objects), implementation of virtual methods (virtual method tables), lifetime of objects (allocating and initializing objects (statically, on the stack, on the heap), constructors, calling inherited constructors, freeing objects, explicit delete/dispose, garbage collection, automatically freeing objects, shared_ptr/unique_ptr, destructors, finalizers), threads and synchronization (implementing threads, basic synchronization constructs, data types with atomic operations), debugging, exceptions (throwing and catching exceptions (try-catch-finally), working with resources: try-with-resources (Java), RAII (C++), using (C#)). Separate compilation, linking, compiler directives: compilation vs. interpreting, role of linking, JIT.

4. Computer Architecture and Operating Systems

Data representation: encoding and layout in memory, bitwise operations and their usage. Computer organization: von Neumann and Harvard architectures, computer memory, secondary storage, address spaces, input/output devices. Instruction set architectures: computer instructions, high-level programming-language constructs and their representation using computer instructions, basic notion of a shared-memory multi-processor. Operating systems: computer and OS initialization, OS kernel, device drivers, privileged and non-privileged code execution, programming interfaces, OS shell environment, user management. Hardware/software interface: device drivers and driver stack, interrupt and exceptions at CPU, OS, and programming language level. Processes and threads: process and thread contexts, process hierarchy, cooperative and preemptive multitasking, scheduling, thread states, active and passive waiting. Race conditions, critical section, mutual exclusion, synchronization primitives, the notions of deadlock and livelock. File and socket APIs, file descriptors, accessing devices as files, standard input and output and their redirection, interprocess communication via pipes.

2 Degree Plans - General Computer Science

Coordinating Department: Computer Science Institute and Department of Applied Mathematics

Specialization Coordinator: doc. Mgr. Robert Šámal, Ph.D.

The specialization General Computer Science is suitable mainly for students interested in obtaining a solid foundation in computer science and mathematics, and who aim to follow their bachelor studies with a master's programme of study. Students are also well prepared for the job market, too. Taking General Computer Science allows the student to pursue algorithms, optimization, and their guiding principles, and also discrete mathematics.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

2.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NPRX005	**Non-procedural Programming**	5	—	2/2 C+Ex
NOPX048	**Linear programming and combinatorial optimization**	5	—	2/2 C+Ex
NMAX055	**Mathematical Analysis 2** [1]	5	2/2 C+Ex	—

[1] In 2019/20 the course Mathematical Analysis 2 is taught in the summer semester for students who started their studies in previous years. Students studying according to the current degree plan will take this course in the winter semester of 2020/21.

2.2 Elective Courses

Elective courses – group 1

A prerequisite for taking either part of the State Final Exam is to have obtained at least 30 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NDMI084	Introduction to approximation and randomized algorithms	5	2/1 C+Ex	—
NDMI098	Algorithmic Game Theory	5	2/2 C+Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMX012	Combinatorics and Graph Theory 2	5	—	2/2 C+Ex
NDMX009	Introduction to Combinatorial and Computational Geometry	6	2/2 C+Ex	—
NOPX046	Discrete and Continuous Optimization	5	—	2/2 C+Ex
NMAX062	Algebra 1	5	2/2 C+Ex	—
NMAI063	Algebra II	3	—	2/0 Ex
NMAX056	Mathematical Analysis 3	5	—	2/2 C+Ex
NMAX042	Numerical Mathematics	5	—	2/2 C+Ex
NMAI059	Probability and Statistics	6	2/2 C+Ex	—
NAIL063	Set Theory	3	—	2/0 Ex

Elective courses – group 2

A prerequisite for taking either part of the State Final Exam is to have obtained at least 5 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX041	Programming in C++	5	2/2 C+Ex	—

NPRX013	Java	5	2/2 C+Ex	—
NPRX035	C# Language and .NET Framework	5	2/2 C+Ex	—

Elective courses – group 3

A prerequisite for taking either part of the State Final Exam is to have obtained at least 45 credits from elective courses overall. There is no specific limit for this third group.

Code	Subject	Credits	Winter	Summer
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR002	Digital Image Processing	4	3/0 Ex	—
NPGR038	Introduction to computer game development	5	—	2/2 C+Ex
NPFL124	Natural Language Processing	4	—	2/1 C+Ex
NPFL012	Introduction to Computer Linguistics	3	2/0 Ex	—
NSWX004	Operating Systems	4	2/1 MC	—
NPRX036	Data Formats	5	—	2/2 C+Ex
NSWI090	Computer Networks	3	—	2/0 Ex
NSWI143	Computer Architecture	3	—	2/0 Ex
NDBI007	Data Organisation and Processing I	4	2/1 C+Ex	—
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NSWI098	Compiler Principles	6	2/2 C+Ex	—
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NSWX142	Web Applications Programming	5	2/2 C+Ex	—
NPRG054	High Performance Software Development	6	—	2/2 C+Ex
NPRX051	Advanced C++ Programming	5	—	2/2 C+Ex
NPRX021	Advanced Programming in Java	5	—	2/2 C+Ex
NPRX038	Advanced C Programming	5	—	2/2 C+Ex

2.3 Recommended Course of Study

The recommended course of study gives all the obligatory courses, while only some elective courses and optional courses are listed. Students need to choose other such courses themselves. Obligatory courses are printed in boldface, elective courses in roman, and optional courses in italics.

First year

Common to all specializations – see under general information above.

Second year

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NMAX055	**Mathematical Analysis 2** [1]	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1** [1]	5	2/2 C+Ex	—
NPRX…	Programming in Java/C#/C++	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NPRX005	**Non-procedural Programming**	5	—	2/2 C+Ex
NOPX048	**Linear programming and combinatorial optimization**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics**	5	2/2 C+Ex	—
NPRG045	**Individual Software Project**	4	—	0/1 C
	Elective course – group 1	5		2/2 C+Ex
	Elective courses			
	Optional courses			

[1] In 2019/20 the courses Mathematical Analysis 2 and Combinatorics and Graph Theory 1 are taught in the summer semester for students who started their studies in previous years. Students studying according to the current degree plan will take these courses in the winter semester of 2020/21.

Third year

Code	Subject	Credits	Winter	Summer
NDBX025	**Database Systems**	5	—	2/2 C+Ex
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
	Elective courses	30		
	Optional courses	15		

Recommended elective courses

To prepare for the State Final Exam, as well as for the further study of computer science, we suggest the following courses in particular.

Code	Subject	Credits	Winter	Summer
NOPX046	Discrete and Continuous Optimization	5	—	2/2 C+Ex
NDMI084	Introduction to approximation and randomized algorithms	5	2/1 C+Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMX009	Introduction to Combinatorial and Computational Geometry	6	2/2 C+Ex	—
NDMX012	Combinatorics and Graph Theory 2	5	—	2/2 C+Ex
NAIL063	Set Theory	3	—	2/0 Ex
NMAX062	Algebra 1	5	2/2 C+Ex	—

2.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the General Computer Science specialization will be further tested according to the list below from topics 1.-3. and from two selected topics among 4.-7. The choice of these two topics is to be declared by the student when signing up for the State Final Exam.

1. Networking Fundamentals

Taxonomy of computer networks. ISO/OSI reference architecture. Overview of the TCP/IP protocol model. Routing. Addresses, ports, sockets. Client-server architectures. Fundamentals of HTTP, FTP and SMTP protocols.

2. Multivariable Differential and Integral Calculus

Riemann integral. Extreme values of multivariable functions. Metric spaces, open and closed sets. Compactness.

3. Combinatorics

Generating functions. Estimates of factorials and binomial coefficients. Ramsey theorems. Error-correcting codes.

4. Optimization Methods

Polyhedra, Minkowski–Weyl theorem. Basics of linear programming, duality theorems, algorithms for LP. Edmonds' algorithm. Integer programming. Approximation algorithms for combinatorial problems (satisfiability, independent set, set cover, scheduling). Applications of linear programming to approximation algorithms. The use of probability in the design of algorithms.

5. Advanced Algorithms and Data Structures

Random-access machine (RAM). Dynamic programming. Strongly connected components of directed graphs. Maximal flows: Dinic and Goldberg algorithms. Application of flows: disjoint paths, matching in bipartite graphs. Flows and paths in graphs with integer weights. Text search algorithms: Knuth–Morris–Pratt, Aho–Corasick, and Rabin–Karp algorithms. DFT and its applications. Approximation algorithms and schemes. Parallel algorithms in Boolean circuits and comparator networks.

6. Geometry

Basic theorems about convex sets (Helly, Rado, separation). Minkowski's lattice theorem. Convex polytopes (basic properties, V-polytopes, H-polytopes, combinatorial complexity). Geometric duality. Voronoi diagrams, hyperplane arrangements, point-line incidences. Elementary computational geometry algorithms (construction of a line arrangement in the plane, construction of a convex hull in the plane).

7. Advanced Discrete Mathematics

Graph colouring (Brooks' and Vizing's theorem). Tutte's theorem. Extremal combinatorics (Turán's theorem, Erdös–Ko–Rado theorem). Drawing graphs on surfaces. Sets and mappings. Subvalence and equivalence of sets. Well-ordered sets. Axiom of choice (Zermelo's theorem, Zorn's lemma).

3 Degree Plans - Programming and Software Development

Coordinating Department: Department of Software Engineering
Specialization Coordinator: RNDr. Filip Zavoral, Ph.D.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

3.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NSWX004	**Operating Systems**	4	2/1 MC	—
NPRX041	**Programming v C++**	5	2/2 C+Ex	—
NSWX142	**Web Applications Programming**	5	2/2 C+Ex	—
NSWI154	**Software Development Tools**	2	0/2 C	—
NSWI041	**Introduction to Software Engineering**	5	—	2/2 C+Ex
NPRX043	**Recommended Programming Practices**	5	—	2/2 MC

3.2 Elective Courses

A prerequisite for taking either part of the State Final Exam is to have obtained at least 37 credits from elective courses overall.

Elective courses – group 1

A prerequisite for taking either part of the State Final Exam is to have obtained at least 5 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX013	Programming in Java	5	2/2 C+Ex	—
NPRX035	Programming in C#	5	2/2 C+Ex	—

Elective courses – group 2

A prerequisite for taking either part of the State Final Exam is to have obtained at least 10 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX051	Advanced C++ Programming	5	—	2/2 C+Ex
NPRX021	Advanced Programming in Java	5	—	2/2 C+Ex
NPRX038	Advanced C# Programming	5	—	2/2 C+Ex
NSWX153	Advanced Web Application Programming	5	—	2/2 C+Ex
NPRX056	Mobile Devices Programming	3	—	0/2 C

Elective courses – group 3

A prerequisite for taking either part of the State Final Exam is to have obtained at least 10 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRG024	Design Patterns	3	—	0/2 MC
NSWI143	Computer Architecture	3	—	2/0 Ex
NPRX036	Data Formats	5	—	2/2 C+Ex
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI090	Computer Networks	3	—	2/0 Ex

Elective courses – group 4

A prerequisite for taking either part of the State Final Exam is to have obtained at least 6 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NSWI098	Compiler Principles	6	2/2 C+Ex	—
NPRG054	High Performance Software Development	6	—	2/2 C+Ex
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex

Elective courses – group 5

Code	Subject	Credits	Winter	Summer
NPGR038	Introduction to computer game development	5	—	2/2 C+Ex
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NDBX007	Database Access Methods	4	2/1 C+Ex	—
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NSWI162	Program semantics	1	0/1 C	—
NSWI163	Introduction to Middleware	1	0/1 MC	—
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NPRX005	Non-procedural Programming	5	—	2/2 C+Ex
NMAX055	Mathematical Analysis 2	5	2/2 C+Ex	—

3.3 Recommended Course of Study

First year

Common to all specializations – see under general information above.

Second year

Code	Subject	Credits	Winter	Summer
NPRX...	Programming in Java/C++/C#	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NSWX142	**web Applications Programming**	5	2/2 C+Ex	—

NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1**	5	2/2 C+Ex	—
NPRX...	Advanced Programming in Java/C++/C#	5	—	2/2 C+Ex
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
	Elective courses			
	Additional courses			

Third year

Code	Subject	Credits	Winter	Summer
NPRX...	Programming in Java/C++/C#	5	2/2 C+Ex	—
	Advanced Programming in Java/C++/C#	5	—	2/2 C+Ex
NSWI098	Compiler Principles [6]	6	2/2 C+Ex	—
NPRG054	High Performance Software Development	6	—	2/2 C+Ex
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
	Elective courses			
	Additional courses			

3.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the Programming and Software Development specialization will be further tested according to the list below.

1. Computer Architecture, Operating Systems and Networks

Runtime memory organization. Memory representation of arrays, structures, and classes. Cache, performance impact, NUMA. Virtual memory. Processing of instructions by processors. Processes and threads, context switching, IRQ, asynchronous I/O. Multicore, multisocket. Physical, link and transport layers. Internetworking, VLAN. TCP/IP adressing.

2. Programming Languages

Object lifetime. Components, dependency injection. Parallel programming, threads. Debugging, exceptions. Reflection and introspection. Principles of dynamic languages. Functional aspects of object languages. Static vs. dynamic polymorphism, generic programming, type deduction. Standard libraries. Design patterns. Principles of WWW, HTTP, URL. Static web pages. Principles of web applications. Client side programming. API of web applications.

3. Software Engineering

Software development processes, requirement analysis, testing, maintenance, risk analysis. Version control systems. Building. Performance measurement. Design of API, classes and methods, object design. Principles of web application security.

4. Databases

Architectures of database systems. Normal forms. Database schema, keys, indexes, integrity constraints. Transactions and their properties. SQL - common statements, subqueries. SQL procedures and functions, triggers.

4 Degree Plans - System Programming

Coordinating Department: Department of Distributed and Dependable Systems

Specialization Coordinator: doc. Ing. Lubomír Bulej, Ph.D.

The System Programming specialization focuses on the prerequisites for the design, development, and maintenance of efficient system software, which provides the foundation for application software. Topics therefore include computer architecture, operating systems, parallel and distributed systems, and middleware. In System Programming emphasis is put on a deeper understanding of modern programming languages used for the development of system software, and on the ability to use modern development tools and approaches.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

4.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NSWX004	**Operating systems**	4	2/1 MC	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NSWI143	**Computer Architecture**	3	—	2/0 Ex
NSWI098	**Compiler Principles**	6	2/2 C+Ex	—
NSWI163	**Introduction to Middleware**	1	0/1 MC	—
NSWI162	**Program semantics**	1	0/1 C	—
NPRG054	**High Performance Software Development**	6	—	2/2 C+Ex
NPRG042	**Programming in Parallel Environment**	6	—	2/2 C+Ex
NSWI090	**Computer Networks**	3	—	2/0 Ex

4.2 Elective Courses

A prerequisite for taking either part of the State Final Exam is to have obtained at least 27 credits from elective courses overall.

Elective courses – Group 1

A prerequisite for taking either part of the State Final Exam is to have obtained at least 5 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX035	Programming in C#	5	2/2 C+Ex	—
NPRX013	Programming in Java	5	2/2 C+Ex	—

Elective courses – Group 2

A prerequisite for taking either part of the State Final Exam is to have obtained at least 5 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX051	Advanced Programming in C++	5	—	2/2 C+Ex
NPRX038	Advanced Programming in C#	5	—	2/2 C+Ex
NPRX021	Advanced Programming in Java	5	—	2/2 C+Ex

Elective courses – Group 3

A prerequisite for taking either part of the State Final Exam is to have obtained at least 7 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NSWI154	Software Development Tools	2	0/2 C	—
NPRX043	Recommended Programming Practices	5	—	2/2 MC
NPRG024	Design Patterns	3	—	0/2 MC
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NSWI054	Software Engineering for Dependable Systems	3	—	0/2 C
NSWI130	Software System Architectures	5	2/2 C+Ex	—

Elective courses – Group 4

A prerequisite for taking either part of the State Final Exam is to have obtained at least 10 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NMAX055	Mathematical Analysis 2	5	2/2 C+Ex	—
NPRX005	Non-procedural Programming	5	—	2/2 C+Ex
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—
NPGR002	Digital Image Processing	4	3/0 Ex	—
NPGR036	Computer Vision	5	—	2/2 C+Ex
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR019	Realtime Graphics on GPU	5	—	2/2 C+Ex
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NPRX037	Microcontroller Programming	5	2/2 C+Ex	—
NPGR038	Introduction to computer game development	5	—	2/2 C+Ex

4.3 Recommended Course of Study

The recommended course of study gives all the obligatory courses, while only some elective courses and optional courses are listed. Students need to choose other such courses themselves. Obligatory courses are printed in boldface, elective courses in roman, and optional courses in italics.

First year

Common to all specializations – see under general information above.

Second year

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NSWX004	**Operating Systems**	4	2/1 MC	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NPRX...	Programming in C#/Java	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NSWI143	**Computer Architecture**	3	—	2/0 Ex
NPRX...	Advanced Programming in C++/C#/Java	5	—	2/2 C+Ex

Third year

Code	Subject	Credits	Winter	Summer
NDMX011	**Combinatorics and Graph Theory 1** [1]	5	2/2 C+Ex	—
NSWI098	**Compiler Principles**	6	2/2 C+Ex	—
NSWI163	**Introduction to Middleware**	1	0/1 MC	—
NSWI162	**Program semantics**	1	0/1 C	—
NPRG045	**Individual Software Project**	4	0/1 C	—
	Elective courses			
	Optional courses			
NSWI090	**Computer Networks**	3	—	2/0 Ex
NPRG054	**High Performance Software Development**	6	—	2/2 C+Ex
NPRG042	**Programming in Parallel Environment**	6	—	2/2 C+Ex
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
	Elective courses			
	Optional courses			

[1] In 2019/2020 the courses Mathematical Analysis 2 and Combinatorics and Graph Theory 1 are taught in the summer semester for students who started their studies in previous years. For students starting their studies in 2019/20, the courses will be taught in the winter semester of 2020/21.

4.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the System Programming specialization will be further tested according to the list below.

1. Computer Architecture

Fundamentals of computer performance (basic metrics and their limitations, relation to program performance). Instruction execution (scalar and superscalar pipelines, static and dynamic instruction scheduling, speculation, hyper-threading, SIMD). Computer memory subsystem architecture (locality of access, caches, cache miss model, cache architectures and their impact on cache performance). Multi-processor systems (impact on computer architecture, cache coherence and its impact on performance, coherence protocols, NUMA). Peripheral device support (communication with devices, interrupts, DMA).

2. Operating Systems

Process and thread management (process and thread state, context switching and its overhead, scheduling on multiprocessors). Communication and synchronization (race condition, synchronization objects and their application in solving synchronization problems, lock implementation, deadlock). Memory management and sharing (address spaces, address translation at HW and SW level, data structures for address translation, fault handling, page sharing, memory mapped files and associated interfaces). File management (concepts and associated interfaces, typical on-disk structures, examples of specific file systems). Device management (interrupts in operating systems, interrupt handling at HW and SW level, device drivers).

3. Computer Networks

Connected and unconnected communication, reliable and unreliable connections. Link layer, addressing, connections at the physical and link layers, VLAN. Network layer, addressing in IPv4 and IPv6, static routing, routers and gateways, NAT, IP tunnels, VPN. Transport layer, addressing and ports in TCP and UDP, reliability, flow control. Service protocls, ARP, DHCP, ICMP. Application interfaces and abstractions, communication security, authentication, encryption.

4. Compilers and Programming Languages

Compiler architecture, AOT and JIT compilation. Lexical and syntactic analysis, internal program representation, basic blocks. Program optimization, automatic vectorization, impact of object-oriented programming on performance. Object lifetime, allocation and initialization (static, stack, heap), constructors, object deallocation (explicit, garbage collection, automatic), destructors, finalizers. Standard libraries and their prominent elements (containers, iterators, algorithms, ranges), common container types (properties and applications), implementing custom containers.

5. Software Design and Construction

API, method, and class design (cohesion, coupling, abstraction, encapsulation, inheritance, composition, polymorphism, immutability). Object-oriented design principles (SOLID) and their applications, application of design principles in design patterns. Parallel programming (speed-up, decomposition, dependencies, load balancing, memory model), atomic operations and non-blocking data structures, design patterns for parallel programs and algorithms. Version management: concepts (version, commit, change set), graph of version, conflict resolution, local/distributed version management. Building software: concepts (artefacts, tasks, dependencies, build targets) and their representation in common build systems. Software testing: unit tests, impact on software design, tools.

5 Degree Plans - Databases and Web

Coordinating Department: Department of Software Engineering
Specialization Coordinator: Prof. RNDr. Tomáš Skopal, Ph.D.

Specialization Databases and Web offers a number of courses on database and web management methods and technologies, data analysis, database administration, programming and development of traditional, web, database and data-intensive applications. Emphasis is put on relational and non-relational databases, social network analysis, web and multimedia search engines, methods of feature extraction. This specialization also gives a solid background in computer science preparing the graduate for subsequent master studies in computer science.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

5.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NSWX142	**Web Applications Programming**	5	2/2 C+Ex	—
NDBI026	**Database Applications**	4	1/2 MC	—
NDBX007	**Database Access Methods**	4	2/1 C+Ex	—
NDBI040	**Modern Database Concepts**	5	2/2 C+Ex	—
NSWX153	**Advanced Programming of Web Applications**	5	—	2/2 C+Ex
NPRX036	**Data Formats**	5	—	2/2 C+Ex
NDBX046	**Data Management**	5	—	2/2 C+Ex
NDBI038	**Searching the web**	4	—	2/1 C+Ex

5.2 Elective Courses

Elective courses – group 1

A prerequisite for taking either part of the State Final Exam is to have obtained at least 15 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX041	Programming in C++	5	2/2 C+Ex	—
NPRX013	Programming in Java	5	2/2 C+Ex	—
NPRX035	Programming in C	5	2/2 C+Ex	—
NPRX051	Advanced C++ Programming	5	—	2/2 C+Ex
NPRX021	Advanced Programming in Java	5	—	2/2 C+Ex
NPRX038	Advanced C Programming	5	—	2/2 C+Ex

NPRX005	Non-procedural Programming	5	—	2/2 C+Ex

Elective courses – group 2

A prerequisite for taking either part of the State Final Exam is to have obtained at least 6 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NSWX004	Operating Systems	4	2/1 MC	—
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NDBI045	Video retrieval and exploration	5	—	2/2 C+Ex
NDBI037	Information Models with Ordering	4	2/1 C+Ex	—
NDBI013	Oracle Administration	2	—	0/2 C

Elective courses – group 3

A prerequisite for taking either part of the State Final Exam is to have obtained at least 3 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NMAX055	Mathematical Analysis 2	5	2/2 C+Ex	—
NPRX056	Mobile Devices Programming	3	—	0/2 C
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI090	Computer Networks	3	—	2/0 Ex
NPGR036	Computer Vision	5	—	2/2 C+Ex
NPGR002	Digital Image Processing	4	3/0 Ex	—
NAIL121	Seminar on Data Mining	4	—	1/2 MC
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—

5.3 Recommended Course of Study

The recommended course of study gives all the obligatory courses, while only some elective courses and optional courses are listed. Students need to choose other such courses themselves. Obligatory courses are printed in boldface, elective courses in roman, and optional courses in italics.

First year

Common to all specializations – see under general information above.

Second year

Code	Subject	Credits	Winter	Summer
NPRG...	Programming in Java/C++/C#	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NSWX142	**Web Applications Programming**	5	2/2 C+Ex	—
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—

NDMX011	**Combinatorics and Graph Theory 1** [1]	5	2/2 C+Ex	—
NSWX153	**Advanced Programming of Web Applications**	5	—	2/2 C+Ex
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
	Elective courses			
	Optional courses			

Third year

Code	Subject	Credits	Winter	Summer
NDBI026	**Database Applications**	4	1/2 MC	—
NDBX007	**Database Access Methods**	4	2/1 C+Ex	—
NDBI040	**Modern Database Concepts**	5	2/2 C+Ex	—
NPRX036	**Data Formats**	5	—	2/2 C+Ex
NDBX046	**Data Management**	5	—	2/2 C+Ex
NDBI038	**Searching the web**	4	—	2/1 C+Ex
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
	Elective courses			
	Optional courses			

[1] In 2019/2020 the courses Mathematical Analysis 2 and Combinatorics and Graph Theory 1 are taught in the summer semester for students who started their studies in previous years. For students starting their studies in 2019/20, the courses will be taught in the winter semester of 2020/21.

5.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the Databases and Web specialization will be further tested according to the list below.

1. Databases

Topics covered by the courses NDBI025 Database Systems, NDBI026 Database Applications, NDBI040 Modern Database Concepts

2. Data Management

Topics covered by the courses NPRX036 Data Formats, NDBX046 Data Management, NDBX007 Database Access Methods

3. Web

Topics covered by the courses NSWI142 Web Applications Programming, NSWX153 Advanced Programming of Web Applications, NDBI038 Searching the Web

6 Degree Plans - Artificial Intelligence

Coordinating Department: Department of Theoretical Computer Science and Mathematical Logic

Specialization Coordinator: Prof. RNDr. Roman Barták, Ph.D.

The specialization Artificial Intelligence integrates the formal foundations of computer science with their application to the solution of complex problems such as automated planning and scheduling, natural language processing, processing of visual, textual and multimedia data, machine learning, big data and data mining, autonomous robotics, and computer vision. The understanding of the mathematical and logical foundations of computer systems given by the specialization Articial Intelligence is directed towards the design of smart systems.

The specialization Artificial Intelligence has the following branches:

– Robotics
– Machine Learning
– Natural Language Processing

Each branch runs according to the same rules, using the same set of obligatory and elective courses, and they have the common topic Foundations of Artificial Intelligence in the State Final Exam. Each branch then has its own additional topic in the State Final Exam.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

6.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NAIL120	**Introduction to Artificial Intelligence**	5	—	2/2 C+Ex
NPRX005	**Non-procedural Programming**	5	—	2/2 C+Ex
NMAX055	**Mathematical Analysis 2**	5	2/2 C+Ex	—

6.2 Elective Courses

Elective courses – group 1

A prerequisite for taking either part of the State Final Exam is to have obtained at least 25 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NPGR002	Digital Image Processing	4	3/0 Ex	—
NPGR036	Computer Vision	5	—	2/2 C+Ex
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—
NAIL121	Seminar on Data Mining	4	—	1/2 MC
NDMI098	Algorithmic Game Theory	5	2/2 C+Ex	—
NPFL012	Introduction to Computer Linguistics	3	2/0 Ex	—
NPFL125	Text processing in UNIX	3	0/2 MC	—
NPFL124	Natural Language Processing	4	—	2/1 C+Ex
NPFL101	Competing in Machine Translation	3	0/2 C	—

NPFL123	Dialogue systems	5	—	2/2 C+Ex
NAIL119	Nature inspired algorithms	5	—	2/2 C+Ex

Elective courses – group 2

A prerequisite for taking either part of the State Final Exam is to have obtained at least 10 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX041	Programming in C++	5	2/2 C+Ex	—
NPRX013	Programming in Java	5	2/2 C+Ex	—
NPRX035	Programming in C#	5	2/2 C+Ex	—

Elective courses – group 3

A prerequisite for taking either part of the State Final Exam is to have obtained at least 10 credits from courses in this group.

Code	Subject	Credits	Winter	Summer
NPRX051	Advanced Programming in C++	5	—	2/2 C+Ex
NPRX021	Advanced Progamming in Java	5	—	2/2 C+Ex
NPRX038	Advanced Programming in C#	5	—	2/2 C+Ex
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRX036	Data Formats	5	—	2/2 C+Ex
NMAX073	Probability and Statistics 2	5	—	2/2 C+Ex
NDBI045	Video retrieval and exploration	5	—	2/2 C+Ex
NOPX046	Discrete and Continuous Optimization	5	—	2/2 C+Ex
NPGR038	Introduction to computer game development	5	—	2/2 C+Ex
NPRX037	Microcontroller Programming	5	2/2 C+Ex	—

6.3 Recommended Course of Study

The recommended course of study gives all the obligatory courses, while only some elective courses and optional courses are listed. Students need to choose other such courses themselves. Obligatory courses are printed in boldface, elective courses in roman, and optional courses in italics.

First year

Common to all specializations – see under general information above.

Second year

Code	Subject	Credits	Winter	Summer
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NTIX061	**Algorithms and Data Structures**	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1** [1]	5	2/2 C+Ex	—

NMAX055	**Mathematical analysis 2** [1]	5	2/2 C+Ex	—
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NPRX041	Programming in C++	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
NPRX051	Advanced Programming in C++	5	—	2/2 C+Ex
NAIL120	**Introduction to Artificial Intelligence**	5	—	2/2 C+Ex
NPRX005	**Non-procedural Programming**	5	—	2/2 C+Ex
	Elective courses			
	Optional courses			

Third year

Code	Subject	Credits	Winter	Summer
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPRX013	Programming in Java	5	2/2 C+Ex	—
NPFL012	Introduction to Computer Linguistics	3	2/0 Ex	—
NPGR036	Computer Vision	5	—	2/2 C+Ex
NAIL121	Seminar on Data Mining	4	—	1/2 MC
NPFL124	Natural Language Processing	4	—	2/1 C+Ex
NPRX036	Data Formats	5	—	2/2 C+Ex
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
	Elective courses			
	Optional courses			

[1] In 2019/2020 the courses Mathematical Analysis 2 and Combinatorics and Graph Theory 1 are taught in the summer semester for students who started their studies in previous years. For students starting their studies in 2019/20, the courses will be taught in the winter semester of 2020/21.

6.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the Artificial Intelligence specialization will be further tested according to the information below.

The topic Foundations of Artificial Intelligence is required in all branches. Each branch then has its own additional topic in the State Final Exam.

Foundations of Artificial Intelligence

Problem solving via search (algorithm A*); constraint satisfaction. Logical reasoning (forward and backward chaining, resolution, SAT); probabilistic reasoning (Bayesian networks); knowledge representation (situation calculus, Markovian models). Automated planning; Markov decision processes. Games and theory of games. Machine learning (decision trees, regression, reinforcement learning).

Recommended courses:

Code	Subject	Credits	Winter	Summer
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex

Branch Robotics

Kinematics: motion and transformation, basic problem-solving. Control systems: architectures, implementation, specific run-time environments. Motion, sensorics: motion types, basic actuators and sensor types, closed loop control, input data processing. Localization and mapping: possibilities for determining position, map types, model situation solving, simultaneous localization and mapping. Image processing, computer vision: object searching and tracking.

Recommended courses:

Code	Subject	Credits	Winter	Summer
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NPGR036	Computer Vision	5	—	2/2 C+Ex
NPRX037	Microcontroller Programming	5	2/2 C+Ex	—

Branch Machine Learning

Supervised learning: classification and regression, error measure, model assessment (test data, cross validation, maximum likelihood), overfitting and regularization, the curse of dimensionality. Instance-based learning, linear and logistic regression, decision trees, pruning, ensemble learning (bagging, boosting, random forest), support vector machines, t-test, chi-squared test. Unsupervised learning, clustering.

Recommended courses:

Code	Subject	Credits	Winter	Summer
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NAIL121	Seminar on Data Mining	4	—	1/2 MC
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—

Branch Natural Language Processing

System of layers in language description, morphological and syntactic analysis. Fundamentals of probability theory and information theory. Statistical methods in natural language processing, language models. Machine learning, classification, regression. Estimation of generalization error, overfitting, regularization. Word embedding, fundamentals of deep learning. Applications in natural language processing, examples of evaluation measures.

Recommended courses:

Code	Subject	Credits	Winter	Summer
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPFL012	Introduction to Computer Linguistics	3	2/0 Ex	—
NPFL124	Natural Language Processing	4	—	2/1 C+Ex

7 Degree Plans - Computer Graphics, Vision and Game Development

Coordinating Department: Department of Software and Computer Science Education

Specialization Coordinator: RNDr. Josef Pelikán

The specialization Computer Graphics, Vision and Game Development is intended for students interested in visual disciplines of computer science – realistic image synthesis, digital image analysis and computer game development. A natural continuation of this specialization is the master degree study plan Visual Computing and Game Development. But this bachelor specialization consists in itself of a complete degree which can be easily utilized in industry:

- Design of graphical systems, e.g. for visual effects or photorealistic visualizations (postproduction teams, visual effect departments, realistic rendering engines, shaders, etc.)
- Employment in any field where digital image processing or computer vision is used (engineering, electronics, development of robotic systems, medicine, security, automatic quality check, remote sensing, etc.)
- Computer game development at multiple levels (game engine developer, GPU programmer /shaders/, game content editing tools, game logic programming, artificial intelligence, data /game levels/ processing, etc.)

The specialization Computer Graphics, Vision and Game Development has the following branches:

- Computer Graphics
- Computer Vision
- Game Development

Each of the three branches follows the same rules and conditions and has the same set of obligatory and elective courses. The first three topics (1. to 3. – see further below) of the State Final Exam requirements are shared among all three branches as well. The only difference is the last specific set of State Final Exam requirements: "profile courses" are offered in order to cover these distinct requirements.

Common obligatory courses in Computer Science

Common obligatory courses for all specializations are listed above in the section giving general information.

7.1 Obligatory Courses

Code	Subject	Credits	Winter	Summer
NPGR003	**Introduction to Computer Graphics**	5	2/2 C+Ex	—
NMAX055	**Mathematical Analysis 2**	5	2/2 C+Ex	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NPRX035	**Programming in C#**	5	2/2 C+Ex	—

7.2 Elective Courses

A prerequisite for taking either part of the State Final Exam is to have obtained at least 43 credits from elective courses overall. Selection of courses should be based on the branch that will be taken in the State Final Exam. Please see the knowledge requirements section below.

Code	Subject	Credits	Winter	Summer
NPGR025	Introduction to Colour Science	3	2/0 Ex	—
NPGR002	Digital Image Processing	4	3/0 Ex	—
NPGR035	Machine learning in computer vision	5	2/2 C+Ex	—
NPGR037	Matlab Practice	3	0/2 C	—
NSWI160	Game Development Middleware	6	0/4 C	—
NSWI159	Practical Course on Game Development [1]	2	0/1 C	0/1 C
NPGR004	Photorealistic Graphics	5	—	2/2 C+Ex
NPGR019	Realtime Graphics on GPU	5	—	2/2 C+Ex
NPGR020	Geometry for Computer Graphics	3	—	2/0 Ex
NPGR036	Computer Vision	5	—	2/2 C+Ex
NPGR038	Introduction to computer game development	5	—	2/2 C+Ex
NSWX004	Operating Systems	4	2/1 MC	—
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NMAX073	Probability and Statistics 2	5	2/2 C+Ex	—
NMAX056	Mathematical Analysis 3	5	—	2/2 C+Ex
NPRX051	Advanced C++ Programming	5	—	2/2 C+Ex
NPRX038	Advanced C# Programming	5	—	2/2 C+Ex
NPRX005	Non-procedural Programming	5	—	2/2 C+Ex
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NPRG054	High Performance Software Development	6	—	2/2 C+Ex
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NMAX042	Numerical Mathematics	5	—	2/2 C+Ex
NDBI045	Video retrieval and exploration	5	—	2/2 C+Ex
NOPX046	Discrete and Continuous Optimization	5	—	2/2 C+Ex

[1] Practical Course on Game Development is taught each semester and students are allowed to take it repeatedly (each successfully completed semester earns two credit points). In fact we recommend students following the branch Game Development to take this course more than once.

7.3 Recommended Course of Study

The recommended course of study gives all the obligatory courses, while only some elective courses and optional courses are listed. Students need to choose other such courses themselves. ”Profile courses” are important because they cover material needed for State Final Exam. The three branches have different profile courses – see the section giving the requirements for the State Final Exam.

There are three recommended courses of study, one for each study branch. Obligatory courses are printed in boldface, elective courses in roman, and optional courses in italics. Profile courses (needed for the State Final Exam) are footnoted 1, generally recommended lectures are footnoted 2, and elective courses without a footnote can be freely substituted by others.

First year

Common to all specializations – see under general information above.

Second year – Computer Graphics

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1**	5	2/2 C+Ex	—
NMAX055	**Mathematical Analysis 2**	5	2/2 C+Ex	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NPRX035	**Programming in C#**	5	2/2 C+Ex	—
NPGR003	**Introduction to Computer Graphics**	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
NPGR004	Photorealistic Graphics [1]	5	—	2/2 C+Ex
NMAX056	Mathematical Analysis 3 [2]	5	—	2/2 C+Ex
NPRX051	Advanced C++ Programming [2]	5	—	2/2 C+Ex
NMAX042	Numerical Mathematics	5	—	2/2 C+Ex

Third year – Computer Graphics

Code	Subject	Credits	Winter	Summer
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NPGR025	Introduction to Colour Science [1]	3	2/0 Ex	—
NPGR002	Digital Image Processing [2]	4	3/0 Ex	—
NMAX073	Probability and Statistics 2 [2]	5	2/2 C+Ex	—
NPGR037	Matlab Practice	3	0/2 C	—
NSWX004	Operating Systems	4	2/1 MC	—
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
NPGR020	Geometry for Computer Graphics [1]	3	—	2/0 Ex
NPGR036	Computer Vision [2]	5	—	2/2 C+Ex
NPRG054	High Performance Software Development	6	—	2/2 C+Ex
NOPX046	Discrete and Continuous Optimization	5	—	2/2 C+Ex
NPGR019	Realtime Graphics on GPU	5	—	2/2 C+Ex

[1] Profile courses (for the given branch) covering material needed for the State Final Exam.
[2] Recommended courses for the given branch (not formally obligatory but strongly recommended). Courses without a footnote may be freely substituted by others.

Second year – Computer Vision

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—
NDMX011	**Combinatorics and Graph Theory 1**	5	2/2 C+Ex	—
NMAX055	**Mathematical Analysis 2**	5	2/2 C+Ex	—
NPRX035	**Programming in C#**	5	2/2 C+Ex	—
NPGR003	**Introduction to Computer Graphics**	5	2/2 C+Ex	—
NPGR002	Digital Image Processing [1]	4	3/0 Ex	—
NPGR037	Matlab Practice [2]	3	0/2 C	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
NPGR036	Computer Vision [1]	5	—	2/2 C+Ex
NPGR020	Geometry for Computer Graphics [2]	3	—	2/0 Ex
NMAX056	Mathematical Analysis 3 [2]	5	—	2/2 C+Ex
NOPX046	Discrete and Continuous Optimization [2]	5	—	2/2 C+Ex

Third year – Computer Vision

Code	Subject	Credits	Winter	Summer
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NPGR035	Machine learning in computer vision [1]	5	2/2 C+Ex	—
NAIX028	Introduction to Robotics [2]	5	2/2 C+Ex	—
NMAX073	Probability and Statistics 2 [2]	5	2/2 C+Ex	—
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
NMAX042	Numerical Mathematics [2]	5	—	2/2 C+Ex
NPGR004	Photorealistic Graphics	5	—	2/2 C+Ex
NPGR019	Realtime Graphics on GPU	5	—	2/2 C+Ex
NPRG054	High Performance Software Development	6	—	2/2 C+Ex

Second year – Game Development

Code	Subject	Credits	Winter	Summer
NTIX061	**Algorithms and Data Structures 2**	5	2/2 C+Ex	—

NDMX011	**Combinatorics and Graph Theory 1**	5	2/2 C+Ex	—
NMAX055	**Mathematical Analysis 2**	5	2/2 C+Ex	—
NPRX035	**Programming in C#**	5	2/2 C+Ex	—
NPRX041	**Programming in C++**	5	2/2 C+Ex	—
NPGR003	**Introduction to Computer Graphics**	5	2/2 C+Ex	—
NTIX071	**Automata and Grammars**	5	—	2/2 C+Ex
NMAX059	**Probability and Statistics 1**	5	—	2/2 C+Ex
NPRG045	**Individual Software Project**	4	—	0/1 C
NPGR019	Realtime Graphics on GPU [1]	5	—	2/2 C+Ex
NPGR038	Introduction to computer game development [1]	5	—	2/2 C+Ex
NPRX038	Advanced C# Programming [2]	5	—	2/2 C+Ex
NPRX051	Advanced C++ Programming [2]	5	—	2/2 C+Ex

Third year – Game Development

Code	Subject	Credits	Winter	Summer
NAIX062	**Propositional and Predicate Logic**	5	2/2 C+Ex	—
NDBX025	**Database Systems**	5	2/2 C+Ex	—
NSWI160	Game Development Middleware [2]	6	0/4 C	—
NSWI159	Practical Course on Game Development [2]	2	0/1 C	0/1 C
NAIX028	Introduction to Robotics	5	2/2 C+Ex	—
NSWX004	Operating Systems	4	2/1 MC	—
NPGR002	Digital Image Processing	4	3/0 Ex	—
NSZZ031	**Bachelor Thesis**	6	—	0/4 C
NPGR020	Geometry for Computer Graphics [1]	3	—	2/0 Ex
NAIL120	Introduction to Artificial Intelligence	5	—	2/2 C+Ex
NPGR004	Photorealistic Graphics	5	—	2/2 C+Ex
NPGR036	Computer Vision	5	—	2/2 C+Ex

7.4 State Final Exam

The State Final Exam knowledge requirements common to all specializations are described in the first section of this chapter (General Information on Computer Science bachelor's degree plans). Students of the Computer Graphics, Vision and Game Development specialization will be further tested according to the information below.

The Computer Graphics, Vision and Game Development specialization includes a further set of requirements shared among its three branches – topics 1. to 3. are covered by obligatory courses NMAI055 Mathematical Analysis 2 and NPGR003 Introduction to Computer Graphics.

1. Mathematical Analysis

Differential and integral calculus of several variables. Riemann integral. Extrema of functions of several variables. Metric space, closed and open sets, compactness.

2. Fundamentals of 2D Computer Graphics

Human visual system and its properties, colours and their perception. Colour representation on computers, colour spaces RGB, CMY, HSV. High dynamic range graphics (HDR). Raster and vector graphics, raster and vector image file formats and their reasonable use. Image transparency (alpha-channel), alpha-blending. Fundamentals of rasterization – converting 2D shapes into raster image. Basic halftoning and dithering used in digital printing.

3. Fundamentals of 3D Computer Graphics

3D coordinate spaces, homogeneous coordinates, matrix transforms. Compound transforms (useful examples), representation of orientation in space. Projection used in 3D scene rendering. GPU realtime rendering: OpenGL basics, GPU architecture, data formats sent to GPU, programming principles (shaders). 3D scene representations, scene hierarchies, visibility algorithms, fundamentals of shading (Phong). Principles of recursive ray tracing.

Knowledge requirements for individual branches

Students following a given individual branch must be prepared for questions from disciplines covered by "profile courses" of the given branch.

Requirements for Computer Graphics

Topics 4. to 6. are covered by the courses NPGR004 Photorealistic Graphics, NPGR025 Introduction to Colour Science and NPGR020 Geometry for Computer Graphics.

4. Photorealistic Graphics

Recursive ray tracing: principles, naive algorithm and its properties. Ray versus 3D scene intersections, speed-up techniques – bounding volumes and bounding hierarchies (BVH), space partitioning, KD-tree, construction of speedup data structures. Reflectance models: empirical methods (Phong), physically more plausible approaches (microfacets). Textures, volumetric textures and modelling of natural phenonema, noise functions. Anti-aliasing in ray-based algorithms, sampling methods, distributed ray tracing. Monte-Carlo in realistic rendering: introduction to photometry, theory of light transport simulation, integral rendering equation. Random walks, Russian roulette, unbiased quadrature using Monte-Carlo, next event estimation (NEE). Combined estimators, path-tracing, bidirectional path-tracing.

5. Colour Science

Fundamental causes of colour, human eye and function of its parts, retina, metamerism. Colour spaces and colour collections, gamuts, colour mixing, colour matching experiments. CIE RGB, sRGB, CIE XYZ, L*a*b and L*u*v. Chromatic adaptation transforms (CAT), white balance operation in digital photography. Colour appearance models (CAM). Examples of colour ordering systems (Munsell, Pantone). Colour measurement devices. Printing technology, laser printers, inkjet printers, offset printing, ICC profiles, PCS, Device ling ICC profile, colour separation.

6. Geometry for Computer Graphics

Plane and solid Euclidean geometry, applications, animation of continuous movement. Quaternions and their application in animation, LERP and SLERP. Projective space and projective mapping, panoramic stitching in photography and scene reconstruction. Double ratio and its use in size readings from images.

Requirements for Computer Vision

Topics 7. to 9. are covered by the courses NPGR002 Digital Image Processing, NPGR036 Computer Vision and NPGR035 Machine Learning in Computer Vision.

7. Digital Image Processing

Image sampling and quantization, Shannon Theorem. Basic manipulations with images. Histogram, contrast stretching, denoising, sharpening. Linear filtering in image and frequency domains. Convolution and Fourier transform. Edge and corner detection. Image degradations and their modelling. Removing basic degradations (motion and out-of-focus blur), inverse and Wiener filter.

8. Computer Vision

Image acquisition, digital image properties. Mathematical morphology. Image segmentation. Image registration and matching. Description of planar objects – basic principles. Invariants for recognition of 2D objects Local feature detection, description and matching. Salient regions. Object detection and tracking (e.g. human faces), optical flow.

9. Machine Learning

Feature selection and preprocessing. Bayesian decision theory, minimum error criterion. Decision trees. Discriminant analysis, linear classifier. Object recognition, supervised classifiers (k-NN, linear, Bayes). Support Vector Machines (SVM). Cluster analysis, iterative and hierarchical methods. Classification quality evaluation.

Requirements for Game Development

Topics 10., 11. and 6. are covered by the courses NPGR038 Introduction to Computer Games Development, NPGR019 Realtime Graphics on GPU and NPGR020 Geometry for Computer Graphics.

10. Computer Games Development

2D games: sprite-based animation, 2D skeleton, parallax scrolling, tilemap systems, pixel art. 3D games: 3D scene, modeling, skeletal animation, rigging. 3D rendering: shaders, shadow casting, particle systems, billboards, screenspace effects. Audio: sound effects, 3D sound, sound engine, sound composition. Design of game architecture, design patterns for computer games. Game design: definition, history, player taxonomy. Introduction to game engine architectures, Unity engine. Management of computer game development, life-cycle of a videogame project.

11. GPU Graphics

Principles of programmable GPU pipeline. Buffers, constant buffers, efficient data streaming to a GPU. Textures and texturing units, purpose of individual shader types. Higher shader languages (GLSL, HLSL). 3D scene lighting (materials, shadows). Stencil buffer and its utilization. Multipass rendering, deferred shading and screen-space effects. Realtime raytracing. GPGPU – massively parallel algorithms on GPU, basics of CUDA/OpenCL.

6. Geometry for Computer Graphics

Plane and solid Euclidean geometry, applications, animation of continuous movement. Quaternions and their application in animation, LERP and SLERP. Projective space and projective mapping, panoramic stitching in photography and scene reconstruction. Double ratio and its use in size readings from images.

Master of Computer Science

Programme coordinator: doc. RNDr. Tomáš Bureš, Ph.D.

1 General Information

The study programme Master of Computer Science consists of the following study branches and their specializations:

- Discrete Models and Algorithms
  - Discrete mathematics and algorithms
  - Geometry and mathematical structures of computer science
  - Optimization
- Theoretical Computer Science
- Software and Data Engineering
  - Software engineering
  - Software development
  - Web engineering
  - Database systems
  - Big data processing
- Software Systems
  - System programming
  - Dependable systems
  - High performance computing
- Computational Linguistics
  - Computational and formal linguistics
  - Statistical methods and machine learning in computational linguistics
- Artificial Intelligence
  - Intelligent agents
  - Machine learning
  - Robotics
- Computer Graphics and Game Development
  - Computer graphics
  - Computer game development

While your study branch has been specified already in your application, the choice of your specialization (when applicable) is up to your later decision (when enrolling for the state final exam).

Assumed knowledge

It is assumed that an incomming student has a suffucient knowledge of mathematics, theoretical computer science, and programming. In particular, students are expected to have a good knowledge of mathematics at the level of our bachelor courses NMAI054 Mathematical Analysis I, NMAI055 Mathematical Analysis II, NMAI059 Probability and Statistics, NMAI062 Algebra I, and NOPT048 Optimization Methods.

Students are also expected to have knowledge equivalent to the courses NDMI002 Discrete Mathematics, NTIN060 Algorithms and Data Structures I, NTIN061 Algorithms and Data Structures II, NTIN071 Automata and Grammars, and NAIL062 Propositional and Predicate Logic. Knowledge from these courses is also expected for the state final exam.

We also expect students to have good knowledge of programming at least at the level of our bachelor courses NPRG030 Programming I, NPRG031 Programming II.

Students missing knowledge in some of the above-mentioned areas are advised to consider taking the relevant bachelor courses in the first year of their Master's studies. Please do not hesitate to contact the programme coordinator in case of doubt.

Software Project

One of the obligatory courses for students of study branches "Software and Data Engineering", "Software Systems", and "Computer Graphics and Game Development" is NPRG023 Software Project, a team software project. For students of study branches "Computational Linguistics" and "Artificial Intelligence" this course is elective and highly recommended. The goal of this course is to practice team work in large software projects lasting typically around 9 months. We strongly suggest that the topic of the project is chosen in accordance with the study branch. It is possible for students of different study branches to work together on a single project, each student contributing based on his/her study branch. The work on the project is finished by public presentation.

The course is coordinated by the *Commission for Software Projects*. After successful presentation of the project, each student will be granted 15 credits. 6 advance credits can be granted to a student by the teacher supervising the project already after the first semester of the project (provided that enough work has been done on the project). These 6 credits are part of the course NPRG027 Credit for Project. The remaining 9 credits will be granted to a student after successful defence of the project as part of the course NPRG023 Software Project. If the student does not request advance credits then credits for both courses will be granted after successful defence by enrolling to the courses together. The commission can grant 3 extra credits to best students as part of the course NPRG028 Extra Credit for Project.

Students can enroll in the courses NPRG023 Software Project and NPRG027 Credit for Project anytime, not just in the usual enrollment period of a given academic year. Nevertheless, each student can try to complete these courses at most twice during the studies.

State Final Exam

The state final exam consists of two parts: a defense of the Master's (diploma) thesis and an oral examination. The student can enroll for each part separately. To finish the studies, both parts of the state final exam must be completed successfully.

Requirements to enroll for the state final exam

- obtaining at least 120 credits,
- passing all obligatory courses of a given branch and specialization,
- obtaining a given number of credits from the elective courses of a given branch and specialization,

– submitting the Master's thesis by the specified deadline (for defence of the Master's thesis).

Master's (diploma) thesis

Students are advised to select the topic of their Master's (diploma) thesis at the end of the first semester. The faculty departments offer many topics for Master's theses each year and students may also suggest their own topics. We recommend selecting the topic of your thesis primarily from those offered by the department that coordinates your study branch. If you prefer a topic offered by another department or your own topic, please consult the coordinator of your study branch.

After the topic is assigned, the student enrolls in the following obligatory courses:

Code	Subject	Credits	Winter	Summer
NSZZ023	**Diploma Thesis I**	6	—	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	—
NSZZ025	**Diploma Thesis III**	15	—	0/10 C

Course credits for these courses are granted by the thesis advisor on the basis of student's work on the thesis. We suggest the students to discuss with their advisor the expected amount of work and the milestones for each of these courses. All three courses can be enrolled in both winter and summer semesters.

Oral examination

The oral part of the state final exam has the same structure for all study branches. The student is examined from two obligatory areas covering the theoretical foundations of computer science (complexity, computability, and data structures) and from three areas specific to a given study branch and selected specialization. The student will select these three examination areas when registering for the final exam. Please note that some study branches have additional obligatory examination areas and some other restrictions might apply.

Note that not all the courses are available in English every year. We recommend students to contact the study branch coordinator for the selected study branch and discuss individual study plans prior the beginnning of each semester.

The obligatory examination areas:

1. Foundations of complexity and computability

Computational models (Turing machines, RAM). Decidable and partially decidable problems. Algorithmically undecidable problems (Halting Problem). Non-deterministic computational model. Basic complexity classes and their relationship. Hierarchy theorems. NP-complete problems, Cook-Levin Theorem. Pseudo-polynomial algorithms, strong NP-completeness. Approximation algorithms and schemes.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN090	Introduction to Complexity and Computability	5	2/1 C+Ex	—

2. Data structures

Search trees ((a,b)-trees, Splay trees). Heaps (regular, binomial). Hashing, handling collisions, universal hashing, selecting a hash function. Analysis of worst-case, amortized-case and expected complexity of data structures. Behavior and analysis of data structures on systems with memory hierarchy.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN066	Data Structures I	5	2/1 C+Ex	—

2 Degree Plans - Discrete Models and Algorithms

Coordinated by: Department of Applied Mathematics
Study branch coordinator: Doc. RNDr. Martin Klazar, Dr.

Specializations:

– Discrete mathematics and algorithms
– Geometry and mathematical structures of computer science
– Optimization

The study branch Discrete Models and Algorithms provides education in the area of discrete (meaning non-continuous) mathematical structures used in Computer Science. It deals also with combinatorial (and other) algorithms and with modeling phenomena and processes by means of such structures and algorithms. In the specialization Optimization it puts emphasis on solid grasp of various kinds of optimization. The study branch enables to its graduates to be in contact with current scientific results and ideally it prepares them for independent research activity.

The graduate knows in depth discrete mathematics and discrete structures used in computer science and can model, using algorithms, various phenomena and processes. According to chosen specialization the graduate has advanced knowledge in one or more of the areas: combinatorics and graph theory, random techniques and methods in discrete mathematics and algorithms, algebraic and topological methods, and finally optimization of various kinds. The graduate can use this knowledge in research when solving difficult theoretical and practical questions in the area of applied mathematics and computer science, in technical and economical practice, and in interdisciplinary research. The graduate can work in research and development in either academia or industry in any position requiring logical reasoning, analytical capabilities, an algorithmic approach, and the exploitation of modern methods of computer science.

2.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NMAI064	**Mathematical Structures**	6	—	2/2 C+Ex
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

2.2 Elective courses - Set 1

The student needs to obtain at least 45 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NAIL076	Logic Programming I	3	2/0 Ex	—
NDMI009	Combinatorial and Computational Geometry I	6	2/2 C+Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMI013	Combinatorial and Computational Geometry II	6	—	2/2 C+Ex
NDMI015	Combinatorial Counting	3	—	2/0 Ex
NDMI018	Approximation and Online Algorithms	6	—	2/2 C+Ex
NDMI025	Randomized Algorithms	6	—	2/2 C+Ex
NDMI028	Linear Algebra Applications in Combinatorics	6	2/2 C+Ex	—
NDMI036	Combinatorial Structures	3	—	2/0 Ex
NDMI037	Geometric Representations of Graphs I	3	2/0 Ex	—
NDMI045	Analytic and Combinatorial Number Theory	3	—	2/0 Ex
NDMI055	Selected Chapters on Combinatorics I	3	2/0 Ex	—
NDMI056	Selected Chapters on Combinatorics II	3	—	2/0 Ex
NDMI059	Graph Minors and Tree Decompositions	3	2/0 Ex	—
NDMI060	Coloring of Graphs and Other Combinatorial Structures	3	2/0 Ex	—
NDMI064	Applied Discrete Mathematics	3	2/0 Ex	—
NDMI065	Matroid Theory	6	—	2/2 C+Ex
NDMI066	Algebraic Number Theory and Combinatorics	3	2/0 Ex	—
NDMI067	Flows, Paths and Cuts	3	2/0 Ex	—
NDMI073	Combinatorics and Graph Theory III	6	2/2 C+Ex	—
NDMI074	Algorithms and Their Implementation	6	—	2/2 C+Ex
NDMI088	Graph Algorithms II	3	—	2/0 Ex
NMAG337	Introduction to Group Theory	5	2/2 C+Ex	—
NMAI040	Introduction to Number Theory	3	2/0 Ex	—
NMAI065	Fundamentals of Category Theory for Computer Scientists	3	2/0 Ex	—
NMAI066	Topological and Algebraic Methods	3	—	2/0 Ex
NMAI067	Logic in Computer Science	3	2/0 Ex	—

NMAI071	Math++	6	—	2/2 C+Ex
NMMA901	Introduction to Complex Analysis (O)	5	2/2 C+Ex	—
NMMA903	Measure and Integration Theory (O)	8	4/2 C+Ex	—
NMMA931	Introduction to Functional Analysis (O)	8	4/2 C+Ex	—
NOPT008	Nonlinear Optimisation Algorithms	6	—	2/2 C+Ex
NOPT016	Integer Programming	6	—	2/2 C+Ex
NOPT017	Multiobjective Optimisation	3	—	2/0 Ex
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—
NOPT034	Mathematical Programming and Polyhedral Combinatorics	5	2/1 C+Ex	—
NOPT042	Constraint Programming	6	2/2 C+Ex	—
NOPT051	Interval Methods	6	2/2 C+Ex	—
NTIN017	Parallel Algorithms	3	—	2/0 Ex
NTIN022	Probabilistic Techniques	6	2/2 C+Ex	—
NTIN063	Complexity	5	—	2/1 C+Ex
NTIN064	Computability	3	—	2/0 Ex
NTIN067	Data Structures II	3	—	2/0 Ex
NTIN103	Introduction to Parameterized Algorithms	6	2/2 C+Ex	—

2.3 Elective courses - Set 2

The student needs to obtain at least 6 credits for the courses from the following set[1]:

Code	Subject	Credits	Winter	Summer
NDMI073	Combinatorics and Graph Theory III	6	2/2 C+Ex	—
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—

[1] For specializations Discrete mathematics and algorithms, Geometry and mathematical structures of computer science we recommend the course NDMI073; for the specialization Optimization we recommend the course NOPT018. After completing one course from Set 2, the credits are counted for that set and the minimal credit requirement for Set 2 is satisfied. If the student completes both courses from Set 2, the credits for the second course are counted for Set 1.

2.4 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, the student will select three examination areas from the following lists. At least two examination areas must be selected from a chosen specialization, one area may be selected from another specialization. In total, each student will get five questions.

a) Specialization ***Discrete mathematics and algorithms***

Examination areas

1. Combinatorics and graph theory
2. Probabilistic methods and combinatorial enumeration
3. Combinatorial optimization

Knowledge requirements

1. Combinatorics and graph theory

Graph colorings (and variants - choosability), graph minors, tree width and its relation to complexity, geometric representations of graphs (characterization theorems, recognizing algorithms), algebraic properties of graphs, matching theory, Ramsey theory and Szemeredi's regularity lemma, set systems (Steiner triple systems, finite geometries).

Recommended courses

Code	Subject	Credits	Winter	Summer
NDMI037	Geometric Representations of Graphs I	3	2/0 Ex	—
NDMI059	Graph Minors and Tree Decompositions	3	2/0 Ex	—
NDMI060	Coloring of Graphs and Other Combinatorial Structures	3	2/0 Ex	—
NDMI073	Combinatorics and Graph Theory III	6	2/2 C+Ex	—

2. Probabilistic methods and combinatorial enumeration

Combinatorial counting, generating functions, recurrences, basic probabilistic models, linearity of expectation, variance and its uses, Markov's inequality and its applications to particular examples, Chernov's inequality, Lovasz local lemma, asymptotic estimates of functions, probabilistic constructions and algorithms.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDMI015	Combinatorial Counting	3	—	2/0 Ex
NDMI025	Randomized Algorithms	6	—	2/2 C+Ex
NTIN022	Probabilistic Techniques	6	2/2 C+Ex	—

3. Combinatorial optimization

Graph algorithms, theory of polyhedra, travelling salesman problem, classes of special matrices, integrality, matchings and flows in networks, matroid theory, ellipsoid method.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN090	Introduction to Complexity and Computability	5	2/1 C+Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMI065	Matroid Theory	6	—	2/2 C+Ex

NOPT034	Mathematical Programming and Polyhedral Combinatorics	5	2/1 C+Ex	—
NDMI088	Graph Algorithms II	3	—	2/0 Ex

b) Specialization ***Geometry and mathematical structures of computer science***

Examination areas

1. Combinatorial and computational geometry
2. Algebraic and topological structures in Computer Science
3. Category theory in Computer Science
4. Number theory in Computer Science

Knowledge requirements

1. Combinatorial and computational geometry

Geometric problems in finite-dimensional spaces, combinatorial properties of geometric configurations, algorithmic applications, design of geometric algorithms, geometric representations of graphs.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDMI009	Combinatorial and Computational Geometry I	6	2/2 C+Ex	—
NDMI013	Combinatorial and Computational Geometry II	6	—	2/2 C+Ex

2. Algebraic and topological structures in Computer Science

Posets (partially ordered sets), suprema and infima, semilattices and lattices. Fix point theorems. Special ordered structures in Computer Science (DCPO, domains). Fundamentals of general topology, topological constructions. Topological approaches in Computer Science (Scott's topology, continuous lattices). Categories of topological spaces and certain kinds of posets used in Computer Science.

Recommended courses

Code	Subject	Credits	Winter	Summer
NMAI064	Mathematical Structures	6	—	2/2 C+Ex
NMAI066	Topological and Algebraic Methods	3	—	2/0 Ex

3. Category theory in Computer Science

Categories, functors, transformations, examples. Limits and colimits, special constructions. Adjunction, relation to categorical constructions. Reflections and coreflections. Examples of adjoint situations. Cartesian closed categories. Categories and structures, especially structures used in Computer Science. Monadic algebras.

Recommended courses

Code	Subject	Credits	Winter	Summer
NMAI065	Fundamentals of Category Theory for Computer Scientists	3	2/0 Ex	—

4. Number theory in Computer Science

Diophantine approximation (Dirichlet's theorem, Farey fractions transcendental numbers). Diophantine equations (Pell's equation, Thue equation, four squares theorem, Hilbert's tenth problem). Prime numbers (bounds on the prime number function, Dirichlet's theorem). Geometry of numbers (lattices, Minkowski's theorem). Congruences (quadratic residues). Integer partitions (identities, e.g., the pentagonal identity).

Recommended courses

Code	Subject	Credits	Winter	Summer
NMAI040	Introduction to Number Theory	3	2/0 Ex	—

c) Specialization ***Optimisation***

Examination areas

1. Nonlinear programming
2. Discrete optimisation processes
3. Multiobjective and integer programming
4. Parametric programming and interval methods

Knowledge requirements

1. Nonlinear programming

Properties of convex sets and convex functions. Generalizations of convex functions. Necessary and sufficient optimality conditions for free and constrained extrema in problems of nonlinear programming. Quadratic programming. Semidefinite programming. Duality in nonlinear programming. Methods for solving problems on free and constrained extrema, including penalization and barrier methods. One-dimensional optimization.

Recommended courses

Code	Subject	Credits	Winter	Summer
NOPT008	Nonlinear Optimisation Algorithms	6	—	2/2 C+Ex
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—

2. Discrete optimisation processes

Algorithmic game theory, election mechanisms, electronic auctions, application of submodular functions in economy. Optimization based on enumeration, generating functions of edge cuts and perfect matchings, enumerative dualities, the maximum cut problem for graphs on surfaces.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDMI064	Applied Discrete Mathematics	3	2/0 Ex	—
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—

3. Multiobjective and integer programming

Various approaches to solving problems with several criteria. Functional associated to a problem of vector programming. Pareto optimal solution. Problems of linear and nonlinear vector optimization. Methods for obtaining Pareto optimal solutions. Problems of linear programming with integrality conditions or with binary variables. Nonlinear optimization problems with integrality conditions.

Recommended courses

Code	Subject	Credits	Winter	Summer
NOPT016	Integer Programming	6	—	2/2 C+Ex
NOPT017	Multiobjective Optimisation	3	—	2/0 Ex
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—

4. Parametric programming and interval methods

Domains of stability of solutions. Domains of solvability. Solvability function for one-parametric and multi-parametric programming. Interval linear algebra (systems of linear equations, regularity, eigenvalues). Linear programming with imprecise data. Deterministic global optimization, lower and upper bounds on objective function and optimum value.

Recommended courses

Code	Subject	Credits	Winter	Summer
NOPT017	Multiobjective Optimisation	3	—	2/0 Ex
NOPT018	Fundamentals of Nonlinear Optimization	6	2/2 C+Ex	—
NOPT051	Interval Methods	6	2/2 C+Ex	—

3 Degree Plans - Theoretical Computer Science

Coordinated by: Department of Theoretical Computer Science and Mathematical Logic; Computer Science Institute of Charles University
Study branch coordinator: Doc. Mgr. Michal Koucký, Ph.D.

This study branch has no specializations.

Theoretical Computer Science provides comprehensive education in theoretical aspects of computational models, algorithm and data structure design, and structural properties of Boolean functions. Students gain understanding of the state-of-the-art techniques in the design of efficient algorithms and data structures, and also learn the limits and possibilities for solving algorithmic problems. In addition to that students acquire mathematical tools necessary to analyze and model algorithmic processes. Students can utilize gained knowledge in practical setting or they can continue by a doctoral study in theoretical computer science or related areas.

The graduate thoroughly understands the limits and possibilities of computational systems, has a broad overview of algorithmic techniques, and is able to apply these techniques to new problems. He also has skills necessary to convey abstract ideas with precision and clarity. The graduate can apply his skills in the design and analysis of complex systems and in the development of innovative solutions and transformative technologies. The graduate is also well prepared for doctoral studies in theoretical computer science and related areas.

3.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NTIN022	**Probabilistic Techniques**	6	2/2 C+Ex	—
NTIN063	**Complexity**	5	—	2/1 C+Ex
NTIN100	**Introduction to Information Transmission and Processing**	5	—	2/1 C+Ex
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

3.2 Elective courses

The student needs to obtain at least 45 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NAIL021	Boolean Functions and Their Applications	3	2/0 Ex	—
NAIL031	Representations of Boolean Functions	3	—	2/0 Ex
NAIL094	Decision Procedures and Verification	6	—	2/2 C+Ex
NMAG563	Introduction to complexity of CSP	3	2/0 Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMI013	Combinatorial and Computational Geometry II	6	—	2/2 C+Ex
NDMI018	Approximation and Online Algorithms	6	—	2/2 C+Ex
NDMI025	Randomized Algorithms	6	—	2/2 C+Ex
NDMI067	Flows, Paths and Cuts	3	2/0 Ex	—
NDMI074	Algorithms and Their Implementation	6	—	2/2 C+Ex
NDMI077	Algorithms for Specific Graph Classes	3	—	2/0 Ex
NDMI088	Graph Algorithms II	3	—	2/0 Ex
NMAG446	Logic and Complexity	3	—	2/0 Ex
NMAG536	Proof Complexity and the P vs. NP Problem	3	—	2/0 Ex
NMAI067	Logic in Computer Science	3	2/0 Ex	—
NOPT034	Mathematical Programming and Polyhedral Combinatorics	5	2/1 C+Ex	—
NSWI072	Data Compression Algorithms	3	2/0 Ex	—
NTIN017	Parallel Algorithms	3	—	2/0 Ex

NTIN018	Probabilistic Analysis of Algorithms	3	2/0 Ex	—
NTIN033	Experimental Analysis of Algorithms	6	—	2/2 C+Ex
NTIN064	Computability	3	—	2/0 Ex
NTIN067	Data Structures II	3	—	2/0 Ex
NTIN073	Recursion	3	2/0 Ex	—
NTIN081	Structural Complexity	3	—	2/0 Ex
NTIN082	Computational Complexity	3	—	2/0 Ex
NTIN084	Bioinformatics Algorithms	6	2/2 C+Ex	—
NTIN085	Selected Topics in Computational Complexity I	5	2/1 C+Ex	—
NTIN086	Selected Topics in Computational Complexity II	5	—	2/1 C+Ex
NTIN087	String Algorithms	3	2/0 Ex	—
NTIN088	Algorithmic Randomness	3	—	2/0 Ex
NTIN096	Pseudo-Boolean Optimization	3	—	2/0 Ex
NTIN097	Hypercube Problems	3	2/0 Ex	—
NTIN098	Advanced Data Structures	3	2/0 Ex	—
NTIN099	Algorithmic Aspects of Boolean Functions and Parameterized Complexity	3	—	2/0 Ex
NTIN104	Foundations of theoretical cryptography	5	—	2/1 C+Ex
NTIN103	Introduction to Parameterized Algorithms	6	2/2 C+Ex	—

3.3 Other recommended courses

Code	Subject	Credits	Winter	Summer
NOPT016	Integer Programming	6	—	2/2 C+Ex
NOPT042	Constraint Programming	6	2/2 C+Ex	—
NTIN023	Dynamic Graph Data Structures	3	2/0 Ex	—

3.4 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, the student will select three other examination areas. Two of them must be from the following list of examination areas; the last one can be either from the following list as well, or it can be any area from the study branch Discrete Models and Algorithms, any area from the specialization Intelligent agents or the specialization Machine learning of the study branch Artificial Intelligence, or any area from the specialization Computer graphics of the study branch Computer Graphics and Game Development. In total, each student will get five questions from the five examination areas.

Examination areas

1. Complexity and computability
2. Boolean functions

3. Algorithms
4. Advanced data structures

Knowledge requirements

1. Complexity and computability

Oracle computation and relativized complexity classes. Polynomial hierarchy. Non-uniform models of computation. Probabilistic complexity classes. Interactive protocols, PCP Theorem. One-way functions and pseudo-random generators. Communication complexity. Proof complexity. Relationships and separations among complexity classes. Recursion theorems and their application. Effectively inseparable sets. Relativized computability and the jump operation. Arithmetic hierarchy.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN063	Complexity	5	—	2/1 C+Ex
NTIN081	Structural Complexity	3	—	2/0 Ex
NTIN082	Computational Complexity	3	—	2/0 Ex
NMAG536	Proof Complexity and the P vs. NP Problem	3	—	2/0 Ex
NTIN064	Computability	3	—	2/0 Ex
NTIN100	Introduction to Information Transmission and Processing	5	—	2/1 C+Ex

2. Boolean functions

Resolution and its completeness. Classes of Boolean functions with special properties. Algorithms for SAT and MAXSAT. Representing Boolean functions using BDD's and OBDD's. SAT solvers and their use for the SMT Problem. Parameterized complexity. Hypercube graphs.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN099	Algorithmic Aspects of Boolean Functions and Parameterized Complexity	3	—	2/0 Ex
NAIL094	Decision Procedures and Verification	6	—	2/2 C+Ex
NTIN097	Hypercube Problems	3	2/0 Ex	—
NAIL021	Boolean Functions and Their Applications	3	2/0 Ex	—
NAIL031	Representations of Boolean Functions	3	—	2/0 Ex

3. Algorithms

Advanced graph algorithms, flows in graphs, algorithms for planar graphs. Linear and semidefinitive programming, polynomial algorithms, applications in graph and approximation algorithms. Combinatorial approximation algorithms and schemes. Probabilistic algorithms, approximate counting, hashing and its applications. Interactive protocols and verification, PCP Theorem and its applications. Parallel models of computation and complexity classes, techniques and examples of parallel algorithms. String algorithms, sequences, sub-sequences, regular expressions and search.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDMI010	Graph Algorithms	3	2/0 Ex	—
NDMI018	Approximation and Online Algorithms	6	—	2/2 C+Ex
NDMI025	Randomized Algorithms	6	—	2/2 C+Ex
NTIN017	Parallel Algorithms	3	—	2/0 Ex
NTIN087	String Algorithms	3	2/0 Ex	—

4. Advanced data structures

Entropy and information. Error-correcting codes. Data compression. Data structures for string processing. Dynamic data structures for graphs. Dictionary data structures. Probabilistic search data structures. Advanced heaps. Data structures for storing integers. Persistent data structures. Self-adjusting data structures. Cache oblivious analysis and optimal algorithms. Data-streaming algorithms.

Recommended courses

Code	Subject	Credits	Winter	Summer
NTIN100	Introduction to Information Transmission and Processing	5	—	2/1 C+Ex
NTIN067	Data Structures II	3	—	2/0 Ex
NTIN087	String Algorithms	3	2/0 Ex	—
NDMI010	Graph Algorithms	3	2/0 Ex	—
NTIN098	Advanced Data Structures	3	2/0 Ex	—
NSWI072	Data Compression Algorithms	3	2/0 Ex	—

4 Degree Plans - Software and Data Engineering

Coordinated by: Department of Software Engineering
Study branch coordinator: Doc. RNDr. Tomáš Skopal, Ph.D.

Specializations:

– Software engineering
– Software development
– Web engineering
– Database systems
– Big data processing

The study branch Software and data engineering aims at expertise in analysis, design and development of complex software solutions, and systems focused on big data processing. The portfolio of courses provided in the study covers a number of technological platforms, from classic, web-based, to modern cloud and distributed solutions. A required part of the study is a work on large software project where students apply not only the theoretical knowledge and technological skills but also team work abilities.

The graduate gains a deep knowledge of software and data engineering based on her/his specialization. With the specialization Software engineering the graduate is able to analyse requirements for software solutions, to design architectures, and to lead the development process. The specialization Software development prepares the graduate for leading a team of SW developers. The development of internet applications is covered by the specialization Web engineering. The graduate of Database systems is able to design schemas of databases and to implement complex database applications. With the Big data processing specialization the graduate is prepared for the role of data scientist with abilities in data mining and related data analytics knowledge.

4.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NPRG027	**Credit for Project**	6	0/4 C	0/4 C
NPRG023	**Software Project**	9	0/6 C	0/6 C
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

4.2 Elective courses

The student needs to obtain at least 48 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NDBI001	Query Languages I	5	2/2 C+Ex	—
NDBI006	Query Languages II	5	—	2/2 C+Ex
NDBI016	Transactions	3	—	2/0 Ex
NDBI019	Stochastic Methods in Databases	3	—	2/0 Ex
NDBI021	Customer preferences	5	—	2/2 C+Ex
NDBI023	Data Mining	9	—	4/2 C+Ex
NDBI034	Multimedia Retrieval	4	2/1 C+Ex	—
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NDBI042	Data Visualization Techniques	3	—	2/1 C+Ex
NMAI060	Probabilistic Methods	3	2/0 Ex	—
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—
NPRG024	Design Patterns	3	—	0/2 MC
NPRG039	Advanced aspects and new trends in XML	5	2/1 C+Ex	—
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex

NPRG054	High Performance Software Development	5	—	2/2 C+Ex
NPRG059	Advanced Programming Praxis	2	0/1 C	—
NSWI021	Computer Networks II	3	—	2/0 Ex
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NSWI045	TCP/IP Protocol Suite	3	—	2/0 Ex
NSWI068	Object and Component Systems	5	2/2 C+Ex	—
NSWI072	Data Compression Algorithms	3	2/0 Ex	—
NSWI073	Modern Networking Solutions	3	0/2 C	—
NSWI080	Middleware	5	—	2/1 C+Ex
NSWI101	System Behaviour Models and Verification	6	2/2 C+Ex	—
NSWI108	Web Semantization	5	2/2 C+Ex	—
NSWI126	Advanced Tools for Software Development and Monitoring	3	—	0/2 C
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI131	Performance Evaluation of Computer Systems	5	—	2/1 C+Ex
NSWI144	Data Integration and Quality	4	2/1 C+Ex	—
NSWI145	Web Services	5	—	2/2 C+Ex
NSWI149	Software Engineering in Practice	3	—	2/0 C
NSWI150	Virtualization and Cloud Computing	3	2/0 Ex	—
NSWI152	Cloud Application Development	3	—	0/2 C
NSWI153	Advanced Technologies for Web Applications	5	—	2/2 C+Ex
NTIN043	Formal Foundations of Software Engineering	5	2/2 C+Ex	—
NTIN067	Data Structures II	3	—	2/0 Ex
NPFL114	Deep Learning	7	—	3/2 C+Ex

4.3 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, there are two additional obligatory areas based on the selected specialization as indicated below. The student will select the last examination area from the following list. In total, each student will get five questions.

Examination areas

1. Software analysis and architectures (obligatory for the specialization Software engineering)
2. Advanced programming (obligatory for the specializations Software engineering and Software development)
3. Software technologies (obligatory for the specialization Software development)
4. Web technologies (obligatory for the specialization Web engineering)

5. Databases - formal foundations and query languages (obligatory for the specializations Web engineering and Database systems)
6. Databases - implementation and administration (obligatory for the specialization Database systems)
7. Big and unstructured data processing (obligatory for the specialization Big data processing)
8. Data mining (obligatory for the specialization Big data processing)

Knowledge requirements

1. Software analysis and architectures

SW development processes, development phases. Business processes and their modeling using BPMN. UML and its use for analysis and design of structure and behavior of SW. Design patterns. SW testing, impact and change analysis. SW project planning, cost estimation, levels of project management. Legal aspects of SW, principal legal environment for IT projects. Types of SW architecture. Modeling and documentation of SW architecture. Classification of SW architecture quality attributes, their description using scenarios and tactics. Service oriented architectures. Algebraic methods, many sorted algebras, initial models. Formal principles of the UML language. OCL as a specification language, formal base of specification. Formal principles of RDF and OWL, the description logic.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NTIN043	Formal Foundations of Software Engineering	5	2/2 C+Ex	—

2. Advanced programming

Object concepts of modern languages. Concepts of classless languages. Generic programming and metaprogramming, generics and templates, policies, traits, type inference. Modern constructs of programming languages. Object references and lifetime. Advanced aspects of imperative languages. Exceptions, exception-safe programming. Implementation of object properties, runtime support, calling conventions, garbage collection. Modern language constructs and code performance. Design patterns. Scripting languages, prototype-based languages. Domain-specific languages. Functional programming. Principles of code quality, best practices. Refactoring. Testing, debugging, monitoring.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG059	Advanced Programming Praxis	2	0/1 C	—
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—
NPRG024	Design Patterns	3	—	0/2 MC
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex

NSWI068	Object and Component Systems	5	2/2 C+Ex	—

3. Software technologies

Operating system architectures, process management, memory management, communication and synchronization, parallelism, virtualization, paging. File systems, access rights and security. Portability, multiplatform applications. Testing and monitoring of performance and functionality. Architectures of web applications, server-side and client-side scripting, cooperation with database systems. Architecture of data servers, transactions, performance optimization. Cluster, Grid, and Cloud. IaaS, PaaS, and SaaS. MapReduce. Load balancing, high availability.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI126	Advanced Tools for Software Development and Monitoring	3	—	0/2 C
NSWI153	Advanced Technologies for Web Applications	5	—	2/2 C+Ex
NDBI036	A Practical Approach to Database Systems	4	2/1 C+Ex	—
NSWI150	Virtualization and Cloud Computing	3	2/0 Ex	—

4. Web technologies

Overview of basic web technologies. Network services for web technologies. Web services. Architecture of client-server applications, server-side and client-side scripting, web frameworks. Database systems in web applications, NoSQL databases, multimedia databases. Indexing and document searching, principles of web search engines. Linked Data, integration of semantic data to web pages. Security of information systems in the Internet environment, authentication, authorization, security models, cryptography basics, data security.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI153	Advanced Technologies for Web Applications	5	—	2/2 C+Ex
NSWI145	Web Services	5	—	2/2 C+Ex
NDBI034	Multimedia Retrieval	4	2/1 C+Ex	—
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex

5. Databases - formal foundations and query languages

Relational calculi, relational algebras. Relational completeness. Safe expressions, equivalences of relational query languages. Transitive closure of relation. Semantics of SQL. SQL standards. Object extension of relational data model. SQL/XML. Text databases – Boolean and vector model. Searching in text collections: Boolean and vector indexes, signatures, query result ranking. Querying XML data, similarity of XML documents. Datalog. Deductive databases. Recursion in SQL. Tableau queries.

Preference modeling and querying, top-k algorithms, Fuzzy Datalog, recommending systems. RDF data model, SPARQL query language, similarity search in multimedia databases, metric indexes for similarity search.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDBI001	Query Languages I	5	2/2 C+Ex	—
NDBI006	Query Languages II	5	—	2/2 C+Ex
NDBI021	Customer preferences	5	—	2/2 C+Ex
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NDBI038	Searching the web	4	—	2/1 C+Ex

6. Databases - implementation and administration

Architectures of database systems. Models and properties of transactions: locking protocols, time stamps. Transaction isolation, resource allocation. Distributed transactions. Error/failure recovery, journals. Distribution with horizontal fragmentation, implementation of NoSQL databases, CAP theorem. Indexing relational data. Spatial access methods. Algorithms for implementation of relational operations and aggregation functions. Query evaluation and optimization. Indexing XML data. Query evaluation in XML query languages. Data compression: text models, coding, Huffman coding, arithmetic coding, LZ algorithms, bitmap compression, sparse matrices compression, Burrows-Wheeler transformation.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDBI016	Transactions	3	—	2/0 Ex
NPRG039	Advanced aspects and new trends in XML	5	2/1 C+Ex	—
NSWI072	Data Compression Algorithms	3	2/0 Ex	—
NSWI144	Data Integration and Quality	4	2/1 C+Ex	—
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NTIN066	Data Structures I	5	2/1 C+Ex	—

7. Big and unstructured data processing

Distribution with horizontal fragmentation, implementation of NoSQL databases, CAP theorem. Big Data management - distribution, scalability, replication, transactions. MapReduce. Key-value storages. Column storages. Document storages. Models for fulltext querying - vector, Boolean, probabilistic models. Data compression: text models, coding. Similarity search in multimedia databases, metric indexes for similarity search. Data visualization techniques. RDF(S) models, description and dynamic logic, web query languages, model of web semantization.

Recommended coursesy

Code	Subject	Credits	Winter	Summer
NDBI043	Text Search Techniques	3	—	2/0 Ex
NDBI040	Modern Database Concepts	5	2/2 C+Ex	—
NDBI034	Multimedia Retrieval	4	2/1 C+Ex	—
NDBI042	Data Visualization Techniques	3	—	2/1 C+Ex

NSWI108	Web Semantization	5	2/2 C+Ex	—

8. Data mining

Basic principles of database systems, data warehouses and OLAP technology. Data mining – data preprocessing, concept description techniques, methods for mining association rules, methods for data classification and prediction, cluster analysis methods, data mining in database systems. Statistical methods for data mining. Discovery of different types of dependencies. Bayesian analysis, bayesian networks. Probabilistic models of information retrieval. Methods of learning for classification and regression. Support Vector Machines and kernel functions. Experiment evaluation. Data visualization techniques.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDBI023	Data Mining	9	—	4/2 C+Ex
NDBI019	Stochastic Methods in Databases	3	—	2/0 Ex
NAIL029	Machine Learning	3	—	2/0 Ex
NDBI042	Data Visualization Techniques	3	—	2/1 C+Ex

5 Degree Plans - Software Systems

Coordinated by: Department of Distributed and Dependable Systems
Study branch coordinator: Prof. Ing. Petr Tůma, Dr.

Specializations:

– System programming
– Dependable systems
– High performance computing

The study branch puts emphasis on system-oriented programming in one of three focus domains. The specialization System programming focuses on coding the basic layers of a computer system (middleware, operating system). In the specialization Dependable systems, the curriculum deals with systematic construction of systems with high reliability, such as embedded and real-time systems. The specialization High performance computing introduces techniques for software development on high performance computing systems (highly parallel systems, distributed systems, clouds). All focus domains pay attention to both the programming tools and methods and the associated architectural knowledge.

The graduate possesses robust programming skills in the given focus domain: System programming for modern operating systems and system-related technologies (middleware, virtual machines), Dependable systems for dealing with the systematic construction of systems with guaranteed reliability, and High performance computing for software development on modern parallel and distributed systems. The graduate has absorbed both the necessary theoretical foundations and the skills required for solving practical programming tasks. He can use modern programming languages and tools. He can adapt to the fast-moving technologies of today and use these technologies in team software projects. He can solve problems individually and systematically, and apply deep system knowledge in delivering outside-the-box solutions.

5.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NMAI060	**Probabilistic Methods**	3	2/0 Ex	—
NPRG043	**Recommended Programming Practices**	6	—	2/2 C+Ex
NPRG059	**Advanced Programming Praxis**	2	0/1 C	—
NSWI126	**Advanced Tools for Software Development and Monitoring**	3	—	0/2 C
NSWI150	**Virtualization and Cloud Computing**	3	2/0 Ex	—
NPRG027	**Credit for Project**	6	0/4 C	0/4 C
NPRG023	**Software Project**	9	0/6 C	0/6 C
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

5.2 Elective courses

The student needs to obtain at least 30 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRG054	High Performance Software Development	5	—	2/2 C+Ex
NPRG056	Developing Applications for Mobile Devices	3	0/2 C	—
NSWE001	Embedded and Real Time Systems	6	—	2/2 C+Ex
NSWI029	Modern Trends in Computer Science	2	0/2 C	0/2 C
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NSWI054	Software Engineering for Dependable Systems	3	—	0/2 MC
NSWI068	Object and Component Systems	5	2/2 C+Ex	—
NSWI080	Middleware	5	—	2/1 C+Ex
NSWI101	System Behaviour Models and Verification	6	2/2 C+Ex	—
NSWI109	Compiler Design	4	—	2/1 C+Ex

NSWI131	Performance Evaluation of Computer Systems	5	—	2/1 C+Ex
NSWI132	Program Analysis and Code Verification	6	—	2/2 C+Ex
NSWI161	Advanced Operating Systems	2	—	0/2 C
NSWI164	Model-driven Development	1	0/1 C	—

5.3 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, the student will get three questions from the following areas based on the chosen specialization. In total, each student will get five questions.

a) Specialization ***System programming***

Examination areas

1. System aspects of computers (SP)
2. Parallel and distributed systems (SP)
3. Modern programming concepts (SP)

Knowledge requirements

1. System aspects of computers (SP)

Parallelism, synchronization and communication. Memory hierarchy. Virtualization support. All topics considered in the context of contemporary parallel architectures.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRG054	High Performance Software Development	5	—	2/2 C+Ex
NSWI161	Advanced Operating Systems	2	—	0/2 C

2. Parallel and distributed systems (SP)

Communication and coordination in distributed environment, algorithms and technologies (specific technologies for RPC, DSM, messaging based on current development).

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NSWI080	Middleware	5	—	2/1 C+Ex

3. Modern programming concepts (SP)

Modern object-oriented concepts, metaprogramming, reflection, aspects. Features of modern programming languages (for example annotations, iterators, generics, lambda functions). Code quality metrics, documentation, refactoring, testing and debugging.

Design patterns. All topics considered in the context of contemporary programming languages.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—

b) Specialization ***Dependable systems***

Examination areas

1. System aspects of computers (DS)
2. Parallel and distributed systems (DS)
3. Formal methods (DS)

Knowledge requirements

1. System aspects of computers (DS)

Real-time and embedded systems (RTES), real-time scheduling, design and modeling of RTES, operating systems for RTES. Mobile devices, design and deployment of mobile applications, resource management.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWE001	Embedded and Real Time Systems	6	—	2/2 C+Ex
NPRG056	Developing Applications for Mobile Devices	3	0/2 C	—

2. Parallel and distributed systems (DS)

Synchronization in concurrent systems. Causality and consensus in distributed environments. Communication, coordination, replication and mobility in distributed environment, algorithms and technologies (specific technologies for RPC, DSM, messaging based on current development).

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NSWI080	Middleware	5	—	2/1 C+Ex

3. Formal methods (DS)

Program behavior modeling, formal tools (timed automata, LTS, Kripke Structure, lattice, LTL, CTL, bisimulation). Contracts and code verification. Static analysis (principles, basic algorithms). Model checking, state explosion, infinite state model checking, compositional reasoning, bounded model checking, CEGAR, symbolic execution, runtime verification, abstract interpretation).

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI101	System Behaviour Models and Verification	6	2/2 C+Ex	—
NSWI132	Program Analysis and Code Verification	6	—	2/2 C+Ex

c) Specialization ***High performance computing***

Examination areas

1. Distributed systems (HPC)
2. Parallel programming (HPC)
3. System aspects of computers (HPC)

Knowledge requirements

1. Distributed systems (HPC)

Causality and consensus in distributed environment. Communication, coordination, replication and mobility in distributed environment, algorithms and technologies (specific technologies for RPC, DSM, messaging based on current development).

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NSWI080	Middleware	5	—	2/1 C+Ex

2. Parallel programming (HPC)

Modern memory architectures (multiprocessor, GPU, NUMA), relationship between data location and access efficiency. Converting computations for data parallelism, parallel solutions of non-homogeneous tasks. Efficient parallel implementation of basic algorithms. Synchronization, load balancing.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRG054	High Performance Software Development	5	—	2/2 C+Ex

3. System aspects of computers (HPC)

Memory hierarchy. Virtualization support. Performance metrics and relevant measurement methods. Performance measurement methodologies. Performance modeling. All topics considered in the context of contemporary parallel architectures.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI131	Performance Evaluation of Computer Systems	5	—	2/1 C+Ex

NSWI150	Virtualization and Cloud Computing	3	2/0 Ex	—

6 Degree Plans - Computational Linguistics

Coordinated by: Institute of Formal and Applied Linguistics
Study branch coordinator: Doc. RNDr. Markéta Lopatková, Ph.D.

Specializations:

- Computational and formal linguistics
- Statistical methods and machine learning in computational linguistics

The aim of the study branch Computational Linguistics is to get the students ready for research in the area of natural language processing and development of applications dealing with both written and spoken language. Examples of such applications are systems of information retrieval, machine translation, grammar checking, text summarization and information extraction, automatic speech recognition, voice control, spoken dialogue systems, and speech synthesis. The emphasis is put on deep understanding of formal foundations and their practical applicability. The study branch Computational Linguistics can be studied in two specializations: (i) computational and formal linguistics, and (ii) statistical methods and machine learning in computational linguistics.

The graduate is familiar with the theoretical foundations of the formal description of natural languages, the mathematical and algorithmic foundations of automatic natural language processing, and state-of-the-art machine learning techniques. Graduates have the ability to apply the knowledge acquired during their studies in the design and development of systems automatically processing natural language and large quantities of both structured and unstructured data, such as information retrieval, question answering, summarization and information extraction, machine translation and speech processing. They are equipped with reasonable knowledge, skills, and experience in software development and teamwork applicable in all areas involving the development of applications aiding human-computer interaction and/or machine learning.

6.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NPFL063	**Introduction to General Linguistics**	5	2/1 C+Ex	—
NPFL067	**Statistical Methods in Natural Language Processing I**	6	2/2 C+Ex	—
NPFL092	**NLP Technology**	5	1/2 MC	—
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

6.2 Elective courses

The student needs to obtain at least 42 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NPFL006	Introduction to Formal Linguistics	3	2/0 Ex	—
NPFL038	Fundamentals of Speech Recognition and Generation	6	2/2 C+Ex	—
NPFL068	Statistical Methods in Natural Language Processing II	6	—	2/2 C+Ex
NPFL070	Language Data Resources	5	—	1/2 MC
NPFL075	Prague Dependency Treebank	6	—	2/2 C+Ex
NPFL079	Algorithms in Speech Recognition	6	—	2/2 C+Ex
NPFL082	Information Structure of Sentences and Discourse Structure	3	—	0/2 C
NPFL083	Linguistic Theory and Grammar Formalisms	6	—	2/2 C+Ex
NPFL087	Statistical Machine Translation	6	—	2/2 C+Ex
NPFL093	NLP Applications	5	—	2/1 MC
NPFL094	Morphological and Syntactic Analysis I	3	2/0 MC	—
NPFL095	Modern Methods in Computational Linguistics	3	0/2 C	—
NPFL096	Computational Morphology	4	—	2/1 Ex
NPFL099	Statistical Dialogue Systems	5	—	2/1 C+Ex
NPFL103	Information Retrieval	6	2/2 C+Ex	—
NPFL104	Machine Learning Methods	5	—	1/2 C+Ex
NPRG027	Credit for Project	6	0/4 C	0/4 C
NPRG023	Software Project	9	0/6 C	0/6 C
NPFL114	Deep Learning	7	—	3/2 C+Ex

6.3 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, there is one obligatory area for this study branch, one obligatory area dependent on the specialization, and one elective examination area. As the last examination area, the student may also select the obligatory area of the other specialization of the study branch Computational Linguistics, or any area from the specialization Intelligent agents or the specialization Machine learning of the study branch Artificial Intelligence, or any area from the specialization Computer graphics of the study branch Computer Graphics and Game Development. In total, each student will get five questions from the five examination areas.

Examination areas

1. Fundamentals of natural language processing (obligatory for both specializations)
2. Linguistic theories and formalisms (obligatory for the specialization Computational and formal linguistics)

3. Statistical methods and machine learning in computational linguistics (obligatory for the specialization Statistical methods and machine learning in computational linguistics)
4. Multimodal technologies and data (elective)
5. Applications in natural language processing (elective)

Knowledge requirements

1. Fundamentals of natural language processing

Fundamentals of general linguistics. System of layers in language description. Dependency syntax, formal definition of dependency trees and their characteristics. The Chomsky hierarchy of languages, context free languages, phrase grammars, unification-based grammars and categorial grammars for a natural language. Design and evaluation of linguistic experiments, evaluation metrics. Basic stochastic methods. Language modeling, basic methods for training stochastic models. Basic algorithms.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPFL067	Statistical Methods in Natural Language Processing I	6	2/2 C+Ex	—
NPFL063	Introduction to General Linguistics	5	2/1 C+Ex	—

2. Linguistic theories and formalisms

Functional Generative Description. Prague Dependency Treebank. Other basic grammar formalisms (Government and Binding, unification-based grammars, feature structures, HPSG, LFG, categorial grammars, (L)TAG). Phonetics, phonology. Computational Morphology. Syntax. Computational lexicography. Topic-focus articulation; information structure, discourse. Coreference. Linguistic typology. Formal grammars and their application in rule-based morphology and parsing.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPFL063	Introduction to General Linguistics	5	2/1 C+Ex	—
NPFL083	Linguistic Theory and Grammar Formalisms	6	—	2/2 C+Ex
NPFL075	Prague Dependency Treebank	6	—	2/2 C+Ex
NPFL094	Morphological and Syntactic Analysis I	3	2/0 MC	—
NPFL006	Introduction to Formal Linguistics	3	2/0 Ex	—

3. Statistical methods and machine learning in computational linguistics

Generative and discriminative models. Supervised machine learning for classification and regression (linear models, other methods: Naive Bayes, decision trees, example-based learning). Support Vector Machines and Kernel functions. Logistic regression. Unsupervised machine learning methods. Bayesian Networks. Bias-variance tradeoff. Language models and noisy channel models. Smoothing, model combination. HMM, trellis, Viterbi, Baum-Welch. Algorithms for statistical tagging. Algorithms for phrase-based and dependency-based statistical parsing.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPFL067	Statistical Methods in Natural Language Processing I	6	2/2 C+Ex	—
NPFL068	Statistical Methods in Natural Language Processing II	6	—	2/2 C+Ex
NPFL104	Machine Learning Methods	5	—	1/2 C+Ex
NPFL087	Statistical Machine Translation	6	—	2/2 C+Ex

4. Multimodal technologies and data

Fundamentals of speech production and perception. Methods of speech signal processing. HMM acoustic modeling of phonemes. The implementation of the Baum-Welch and Viterbi algorithms in speech recognition systems. Continuous speech recognition using large dictionaries. Adaptation techniques. Speech summarization. Topic and keyword spotting in speech corpora. Speaker recognition. Methods of speech synthesis. Text processing for speech synthesis. Prosody modeling. Basic components of a dialog system. Spoken language understanding. Dialog control – MDP and POMDP systems. Reinforcement learning. Dialogue state tracking in MDP and POMDP systems. User simulation. Speech generation. Dialog systems quality evaluation. Search and indexing in audio-visual archives.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPFL038	Fundamentals of Speech Recognition and Generation	6	2/2 C+Ex	—
NPFL079	Algorithms in Speech Recognition	6	—	2/2 C+Ex
NPFL099	Statistical Dialogue Systems	5	—	2/1 C+Ex

5. Applications in natural language processing

Spell-checking and grammar-checking. Input methods. Machine translation. Machine-aided translation. Statistical methods in machine translation. Quality evaluation of machine translation. Information retrieval, models for information retrieval. Query expansion and relevance feedback. Document clustering. Web search. Duplicate detection and plagiarism detection. Information retrieval evaluation. Sentiment analysis, social network analysis. Search systems (Lucene, SOLR, Terrier). NLP toolkits (GATE, NLTK, NLPTools).

Recommended courses

Code	Subject	Credits	Winter	Summer
NPFL087	Statistical Machine Translation	6	—	2/2 C+Ex
NPFL103	Information Retrieval	6	2/2 C+Ex	—
NPFL093	NLP Applications	5	—	2/1 MC

7 Degree Plans - Artificial Intelligence

Coordinated by: Department of Theoretical Computer Science and Mathematical Logic
Study branch coordinator: Prof. RNDr. Roman Barták, Ph.D.

Specializations:

– Intelligent agents
– Machine learning
– Robotics

The study branch Artificial Intelligence provides education in the area of theoretical and applied knowledge for design of intelligent systems in various areas including data analysis, automated problem solving, and robotic applications. The emphasis is put on deep understanding of formal theoretical foundations and their practical applicability. Students will gain knowledge about design of efficient data structures, about formal modeling of problems and knowledge by using techniques of mathematical logic and probability theory, about algorithms (classical and nature-inspired) for problem solving, for control of autonomous agents, for machine learning, and for data mining, and about complexity analysis of computational methods. The students will learn how to apply these techniques and how to extent them both for abstract (data) and physical (robotic) worlds in single-agent and multi-agent environments. The study branch Artificial Intelligence can be studied in three specializations: Intelligent agents, Machine learning, and Robotics.

Graduates can apply and further extend techniques for the design of intelligent systems, including knowledge modeling and formal modeling of complex systems by means of mathematical logic and probability theory, automated problem solving, planning and scheduling, control of autonomous agents (both virtual and physical), machine learning, and data mining. They are also able to analyse and formally model a complex decision problem, propose an appropriate solving technique, and implement it. Graduates can work in research and development in either academia or industry in any position requiring logical reasoning, analytical capabilities, an algorithmic approach, and the exploitation of modern methods of computer science (declarative and nature-inspired programming).

7.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NMAI060	**Probabilistic Methods**	3	2/0 Ex	—
NAIL070	**Artificial Intelligence II**	3	—	2/0 Ex
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

7.2 Elective courses

The student needs to obtain at least 60 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NAIL002	Neural Networks	9	4/2 C+Ex	—
NAIL013	Applications of Neural Networks Theory	3	—	2/0 Ex
NAIL060	Neural Networks Implementation I	6	2/2 C+Ex	—
NAIL021	Boolean Functions and Their Applications	3	2/0 Ex	—
NAIL022	Methods of Logic Programming	3	2/0 Ex	—
NAIL025	Evolutionary Algorithms I	6	2/2 C+Ex	—
NAIL086	Evolutionary Algorithms II	6	—	2/2 C+Ex
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NAIL029	Machine Learning	3	—	2/0 Ex
NAIL004	Seminar on Artificial Intelligence I	3	0/2 C	—
NAIL052	Seminar on Artificial Intelligence II	3	—	0/2 C
NAIL061	Seminar on Mobile Robotics	3	—	0/2 C
NAIL065	Evolutionary Robotics	5	—	2/1 C+Ex
NAIL068	Human-like Artificial Agents	6	—	2/2 C+Ex
NAIL069	Artificial Intelligence I	5	2/1 C+Ex	—
NAIL071	Planning and Scheduling	3	—	2/0 Ex
NAIL073	Robot I	3	0/2 C	—
NAIL074	Robot II	3	—	0/2 C
NAIL076	Logic Programming I	3	2/0 Ex	—
NAIL077	Logic Programming II	3	—	2/0 Ex
NAIL078	Lambda Calculus and Functional Programming I	5	2/1 C+Ex	—
NAIL087	Computers and Cognitive Sciences I	6	3/1 C+Ex	—
NAIL088	Computers and Cognitive Sciences II	6	—	3/1 C+Ex
NAIL094	Decision Procedures and Verification	6	—	2/2 C+Ex
NAIL101	Probabilistic Robotics	6	—	2/2 C+Ex
NAIL104	Probabilistic graphical models	3	2/0 Ex	—
NAIL105	Internet and Classification Methods	3	—	1/1 C+Ex
NAIL106	Multiagent Systems	6	—	2/2 C+Ex
NAIL107	Machine Learning in Bioinformatics	6	—	2/2 C+Ex
NAIL108	Mobile Robotics	3	—	1/1 MC
NAIL116	Social network and their analysis	6	2/2 C+Ex	—
NDBI023	Data Mining	9	—	4/2 C+Ex
NDBI031	Statistical Methods in Data Mining Systems	3	1/1 C+Ex	—
NMAI061	Methods of Mathematical Statistics	5	—	2/1 C+Ex
NMAI067	Logic in Computer Science	3	2/0 Ex	—

NOPT021	Game Theory	3	2/0 Ex	—
NOPT042	Constraint Programming	6	2/2 C+Ex	—
NPFL067	Statistical Methods in Natural Language Processing I	6	2/2 C+Ex	—
NPFL068	Statistical Methods in Natural Language Processing II	6	—	2/2 C+Ex
NPFL097	Selected Problems in Machine Learning	3	0/2 C	—
NPFL104	Machine Learning Methods	5	—	1/2 C+Ex
NPGR001	Computer vision and intelligent robotics	3	2/0 Ex	—
NPRG023	Software Project	9	0/6 C	0/6 C
NPRG027	Credit for Project	6	0/4 C	0/4 C
NPRG037	Microcontroller Programming	6	2/2 C+Ex	—
NSWE001	Embedded and Real Time Systems	6	—	2/2 C+Ex
NSWI054	Software Engineering for Dependable Systems	3	—	0/2 MC
NPFL114	Deep Learning	7	—	3/2 C+Ex

7.3 Other recommended courses

Code	Subject	Credits	Winter	Summer
NAIL015	Neural Networks Implementation II	6	—	2/2 C+Ex
NAIL031	Representations of Boolean Functions	3	—	2/0 Ex
NAIL079	Lambda Calculus and Functional Programming II	5	—	2/1 C+Ex
NAIL110	Practical Course in Robotics	6	—	1/3 MC
NPFL038	Fundamentals of Speech Recognition and Generation	6	2/2 C+Ex	—
NPFL054	Introduction to Machine Learning	5	2/2 C+Ex	—
NPFL079	Algorithms in Speech Recognition	6	—	2/2 C+Ex
NPFL087	Statistical Machine Translation	6	—	2/2 C+Ex
NPFL103	Information Retrieval	6	2/2 C+Ex	—
NPGR002	Digital Image Processing	5	3/0 Ex	—
NTIN084	Bioinformatics Algorithms	6	2/2 C+Ex	—
NSWI103	Project Management – System Dynamics I	3	0/2 C	—
NSWI104	Company Management – System Dynamics II	3	—	0/2 C

7.4 State Final Exam

In addition to the two examination areas that are obligatory for all study branches, the student will select three examination areas from the following lists. At least two examination areas must be selected from a chosen specialization (one area is obligatory for the specialization Intelligent agents), one area may be selected from another specialization. In total, each student will get five questions.

a) Specialization ***Intelligent agents***

Examination areas

1. Knowledge representation (obligatory for the specialization Intelligent agents)
2. Problem solving and planning
3. Nonprocedural programming
4. Multi-agent systems
5. Nature inspired computing

Knowledge requirements

1. Knowledge representation

Formal systems, first-order logic, syntax, axioms, inference rules. Propositional logic, its semantics, tautology and satisfiability, provability. Normal forms of formulae. Automated theorem proving, model checking (DPLL), forward and backward chaining, resolution and unification.

Conditional independence, Bayesian networks, evaluation in Bayesian networks, naive Bayes classifier, decision graphs, MDP, POMDP, reinforcement learning, conditional random fields.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL062	Propositional and Predicate Logic	6	2/2 C+Ex	—
NMAI060	Probabilistic Methods	3	2/0 Ex	—
NAIL069	Artificial Intelligence I	5	2/1 C+Ex	—
NAIL070	Artificial Intelligence II	3	—	2/0 Ex
NAIL104	Probabilistic graphical models	3	2/0 Ex	—

2. Problem solving and planning

Knowledge representation: state space, production systems, logical representation. Search algorithms: tree, graph, and local search, heuristics. SAT solving and constraint satisfaction. Problem modeling. Automated planning: planning domain and problem, planning operators. Planning techniques and algorithms, heuristics, and extensions of planning techniques. Planning and scheduling.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL069	Artificial Intelligence I	5	2/1 C+Ex	—
NAIL071	Planning and Scheduling	3	—	2/0 Ex
NOPT042	Constraint Programming	6	2/2 C+Ex	—
NAIL094	Decision Procedures and Verification	6	—	2/2 C+Ex

3. Nonprocedural programming

Differences between procedural and non-procedural styles of programming. Principles of functional and logic programming. Lambda calculus, its syntax, and reduction principles. Church and Rosser property and consistency of calculus. Fixed point theorems. Normal form of terms. Typed lambda calculus.

Substitution and unification. Horn clauses, SLD resolution and logic programs. Pure Prolog, negation as failure, general logic programs. Sufficient conditions of program termination. Implementation of Prolog. Constraint logic programming.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL076	Logic Programming I	3	2/0 Ex	—
NAIL077	Logic Programming II	3	—	2/0 Ex
NAIL078	Lambda Calculus and Functional Programming I	5	2/1 C+Ex	—
NAIL022	Methods of Logic Programming	3	2/0 Ex	—
NOPT042	Constraint Programming	6	2/2 C+Ex	—

4. Multi-agent systems

Autonomous agent architectures; agent perception, agent action selection mechanism, agent memory. Psychological inspiration. Methods for agent control; symbolic and connectionist reactive planning, hybrid approaches. Path search problem, steering rules, terrain representation. Communication and knowledge in multiagent systems, ontologies, speech acts, FIPA-ACL, protocols. Distributed problem solving, cooperation, Nash equilibria, Pareto efficiency, source allocation, auctions. Agent design methodologies, agent languages and environments. Ethological inspiration, models of population dynamics. Methods for agent learning; reinforcement learning, basic forms of animal learning.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL106	Multiagent Systems	6	—	2/2 C+Ex
NAIL068	Human-like Artificial Agents	6	—	2/2 C+Ex

5. Nature inspired computing

Genetic algorithms, genetic and evolutionary programming. Schemata theory, probabilistic models of simple genetic algorithm. Evolutionary strategies, differential evolution, coevolution, open ended evolution. Swarm optimization algorithms. Memetic algorithms, hill climbing, simulated annealing. Application of evolutionary algorithms (expert systems evolution, neuroevolution, combinatorial optimization, multi-objective optimization).

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL025	Evolutionary Algorithms I	6	2/2 C+Ex	—
NAIL086	Evolutionary Algorithms II	6	—	2/2 C+Ex
NAIL065	Evolutionary Robotics	5	—	2/1 C+Ex

b) Specialization ***Machine learning***

Examination areas

1. Machine learning and its applications
2. Neural networks
3. Data mining

Knowledge requirements

1. Machine learning and its applications

Machine learning; version space search, supervised learning and self-organization, probabilistic approaches, reinforcement learning, theoretical aspects of machine learning. Evolutionary algorithms; fundamental concepts and theoretical knowledge, the building block hypothesis, coevolution, applications of evolutionary algorithms. Machine learning in computational linguistics. Probabilistic algorithms for biological sequence analysis; search for motifs in DNA sequences, strategies for gene detection and protein structure prediction.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL029	Machine Learning	3	—	2/0 Ex
NPFL067	Statistical Methods in Natural Language Processing I	6	2/2 C+Ex	—
NAIL025	Evolutionary Algorithms I	6	2/2 C+Ex	—
NAIL107	Machine Learning in Bioinformatics	6	—	2/2 C+Ex

2. Neural networks

Neurophysiological minimum. Models for supervised learning, the back-propagation algorithm, strategies to speed up the training process, regularization techniques and generalization. Associative memories; Hebbian learning and the search for suboptimal solutions, stochastic models. Artificial neural networks based on the principle of unsupervised learning. Modular, hierarchical and hybrid models of neural networks. Genetic algorithms and their application for training of artificial neural networks.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL002	Neural Networks	9	4/2 C+Ex	—
NAIL060	Neural Networks Implementation I	6	2/2 C+Ex	—
NAIL013	Applications of Neural Networks Theory	3	—	2/0 Ex
NAIL065	Evolutionary Robotics	5	—	2/1 C+Ex

3. Data mining

Basic paradigms of data mining. Data preparation, attribute selection and methods for relevance analysis of attributes. Data mining methods; association rules, approaches based on supervised learning and cluster analysis. Methods for the extraction of characteristic and discriminant rules and measures of their interestingness. Representation, evaluation and visualization of the extracted knowledge. Models and evaluation of information retrieval. Practical applications of data mining techniques and methods for information retrieval.

Recommended courses

Code	Subject	Credits	Winter	Summer
NDBI023	Data Mining	9	—	4/2 C+Ex
NPFL103	Information Retrieval	6	2/2 C+Ex	—
NAIL105	Internet and Classification Methods	3	—	1/1 C+Ex
NAIL099	Seminar of machine learning and modelling I	2	0/1 C	—

c) Specialization ***Robotics***

Examination areas

1. Localization and mapping
2. Control systems
3. Robotic systems
4. Planning and navigation

Knowledge requirements

1. Localization and mapping

Basic localization methods. Probabilistic localization, particle filters, Monte-Carlo methods. Environment representation, map formats, correspondence problem, mapping in dynamic environment. Localization and mapping relation, SLAM.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NAIL101	Probabilistic Robotics	6	—	2/2 C+Ex
NAIL108	Mobile Robotics	3	—	1/1 MC

2. Control systems

Robot control systems. Signal processing, object recognition, feature matching and tracking. Modeling systems, virtual robotics, simulators. Distributed algorithms, multirobot control systems, communication, synchronization, coordination. Software implementation, programming for specific runtime environment, debugging tools and techniques.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NPGR001	Computer vision and intelligent robotics	3	2/0 Ex	—
NPGR002	Digital Image Processing	5	3/0 Ex	—
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—

3. Robotic systems

Basic kinematic and dynamic model, inverse kinematics and dynamics. Low-level hardware and software, embedded systems. Sensor and actuator types, principles and typical usage. High-level robot systems and their control: manipulators, mobile robotics, autonomous robotics.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NAIL108	Mobile Robotics	3	—	1/1 MC
NSWE001	Embedded and Real Time Systems	6	—	2/2 C+Ex

4. Planning and navigation

Basic navigation techniques: dead-reckoning, odometry, triangulation, and trilateration, inertial navigation. Navigation and exploration algorithms. Action planning, planning problem formulation, basic planning and planning with time and resources.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NAIL108	Mobile Robotics	3	—	1/1 MC
NAIL071	Planning and Scheduling	3	—	2/0 Ex

8 Degree Plans - Computer Graphics and Game Development

Coordinated by: Department of Software and Computer Science Education
Study branch coordinator: Doc. Alexander Wilkie, Dr.

Specializations:

- Computer graphics
- Computer game development

The study branch consists of two closely related specializations, Computer graphics and Computer game development. The specialization Computer graphics offers training in a wide range of visual sciences, including geometric modeling, rendering (image synthesis) as well as the basics of image analysis and computer vision. The specialization Computer game development focuses - apart from computer graphics techniques - mainly on artificial intelligence and intelligent agent systems, as well as on software engineering skills necessary for the development of large-scale gaming projects. Both specializations place emphasis on general programming skills, both at the system level closer to the hardware, as well as on the higher level of modern programming languages.

Graduates have expertise in the design and development of graphical systems and computer games, but they can work in any position which requires logical thinking, analytic and algorithmic approaches or the use of methods of computer science. Depending on the chosen specialization, graduates have a deep knowledge of computer graphics and image analysis, and their expertise covers the development of large-scale gaming projects, real-time applications, programming of portable devices, as well as the foundations of artificial intelligence and computer graphics in the context of computer games. Graduates can apply this knowledge to solve specific practical tasks. They can work in research and development both in the private sector and in academia.

8.1 Obligatory courses

Code	Subject	Credits	Winter	Summer
NTIN090	**Introduction to Complexity and Computability**	5	2/1 C+Ex	—
NTIN066	**Data Structures I**	5	2/1 C+Ex	—
NMAI060	**Probabilistic Methods**	3	2/0 Ex	—
NPRG027	**Credit for Project**	6	0/4 C	0/4 C
NPRG023	**Software Project**	9	0/6 C	0/6 C
NSZZ023	**Diploma Thesis I**	6	0/4 C	0/4 C
NSZZ024	**Diploma Thesis II**	9	0/6 C	0/6 C
NSZZ025	**Diploma Thesis III**	15	0/10 C	0/10 C

8.2 Elective courses

The student needs to obtain at least 47 credits for the courses from the following set:

Code	Subject	Credits	Winter	Summer
NPGR007	Advanced 2D Computer Graphics	5	2/1 C+Ex	—
NPGR010	Computer Graphics III	6	2/2 C+Ex	—
NPGR001	Computer vision and intelligent robotics	3	2/0 Ex	—
NPGR021	Geometric Modelling	6	2/2 C+Ex	—
NPGR026	Predictive Image Synthesis Technologies	6	—	2/2 C+Ex
NPGR027	Shading Languages	5	—	2/1 C+Ex
NPGR028	High Performance Ray Tracing	3	—	2/0 Ex
NPGR033	Computer Graphics for Game Development	6	—	2/2 C+Ex
NPGR024	Seminar on Scientific Soft Skills	3	—	0/2 C
NPGR029	Variational methods in image processing	3	—	2/0 Ex
NSWI115	Computer Games Development	6	2/2 C+Ex	—
NPRG056	Developing Applications for Mobile Devices	3	0/2 C	—
NSWI035	Principles of Distributed Systems	3	2/0 Ex	—
NPRG054	High Performance Software Development	5	—	2/2 C+Ex
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NAIL069	Artificial Intelligence I	5	2/1 C+Ex	—
NAIL070	Artificial Intelligence II	3	—	2/0 Ex
NPRG059	Advanced Programming Praxis	2	0/1 C	—
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NAIL068	Human-like Artificial Agents	6	—	2/2 C+Ex

NAIL071	Planning and Scheduling	3	—	2/0 Ex
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex
NMAI061	Methods of Mathematical Statistics	5	—	2/1 C+Ex
NAIL106	Multiagent Systems	6	—	2/2 C+Ex
NPGR016	Applied Computational Geometry	5	—	2/1 C+Ex
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—
NPFL114	Deep Learning	7	—	3/2 C+Ex

8.3 Other recommended courses

Code	Subject	Credits	Winter	Summer
NPGR005	Special Seminar for Computer Graphics	2	0/2 C	0/2 C
NPGR013	Special Functions and Transformations in Image Processing	3	—	2/0 Ex
NPGR022	Advanced Seminar On Image Processing	2	0/2 C	0/2 C
NAIL028	Introduction to Robotics	6	2/2 C+Ex	—
NAIL108	Mobile Robotics	3	—	1/1 MC
NSWI072	Data Compression Algorithms	3	2/0 Ex	—
NPGR012	Interactive 3D web graphics	6	2/2 C+Ex	2/2 C+Ex
NPGR030	Optics for computer graphics	3	2/0 Ex	—
NPGR023	Visualization	5	2/1 C+Ex	—
NAIL025	Evolutionary Algorithms I	6	2/2 C+Ex	—
NAIL082	Seminar on Humanlike Artificial Agents	3	0/2 C	0/2 C
NAIL087	Computers and Cognitive Sciences I	6	3/1 C+Ex	—
NPOZ017	Video Games as a Socio-cultural Phenomenon	3	2/0 Ex	—
NSWI149	Software Engineering in Practice	3	—	2/0 C
NPRG058	Advanced Programming in Parallel Environment	3	0/2 C	—
NSWI159	Practical Course on Game Development	2	0/1 C	0/1 C
NSWI160	Game Development Middleware	6	0/4 C	—

8.4 State Final Exam

a) Specialization ***Computer graphics***

In addition to the two examination areas that are obligatory for all study branches, the student will select three other examination areas. Students have to choose at least two examination areas from the following list of examination areas; the third examination area can be chosen at will also from among all the examination areas offered in the specialization Computer game development, except the area Computer graphics for games. Students will be asked one question from each selected examination area. In total, each student will get five questions.

Examination areas

1. Geometric modeling and computational geometry
2. Image analysis and processing, image compression, computer vision
3. Realistic image synthesis

Knowledge requirements

1. Geometric modeling and computational geometry

Homogeneous coordinates, affine and projective transformations in the plane and in space, quaternions in the representation of the 3D orientation, differential geometry of curves and surfaces, spline functions, cubic splines C2, interpolation by cubic splines, Bezier curves, Catmull-Rom splines, B-splines, de Casteljau and de Boor algorithms, approximation surfaces, surfaces defined by the edges, Bezier surfaces, patch stitching, B-spline surfaces, NURBS surfaces. Design of geometric algorithms and their complexity, triangulation of polygons and point sets, Voronoi diagrams and Delaunay triangulations, convex hull, intersections of geometric shapes, dualization, localization, data structures for efficient spatial search.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR016	Applied Computational Geometry	5	—	2/1 C+Ex
NPGR021	Geometric Modelling	6	2/2 C+Ex	—
NPGR007	Advanced 2D Computer Graphics	5	2/1 C+Ex	—

2. Image analysis and processing, image compression, computer vision

Mathematical models of images, 2D Fourier transform and convolution, sampling and quantization of the image, contrast and brightness change, noise removal, edge detection, inverse and Wiener filter, determining the relative position of images - registration, point and object correspondence, geometric distortion removal, edge detection, detection of areas, features for description and recognition of 2D objects, moment invariants, wavelets and their use, statistical theory of pattern recognition, supervised and unsupervised classification, computer vision.

Reproduction of color graphics, halftoning and dithering, alpha blending and image composition, warping, morphing, compression of raster 2D graphics, scalar and vector quantization, predictive compression, transformation compression methods, hierarchical and progressive methods, video compression, temporal prediction (motion compensation), JPEG and MPEG standards, image capture in digital photography.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR002	Digital Image Processing	5	3/0 Ex	—
NPGR029	Variational methods in image processing	3	—	2/0 Ex

NPGR013	Special Functions and Transformations in Image Processing	3	—	2/0 Ex
NPGR007	Advanced 2D Computer Graphics	5	2/1 C+Ex	—
NPGR001	Computer vision and intelligent robotics	3	2/0 Ex	—

3. Realistic image synthesis

Representation of 3D scenes, visibility determination, cast shadows, reflection models and shading algorithms, recursive ray tracing, textures, anti-aliasing, acceleration methods for ray-tracing, principle of radiosity methods, physically-based models of light transport (radiometry, rendering equation), Monte Carlo integration, Monte Carlo approaches in lighting simulation, direct methods in volume data visualization, isosurface extraction. Graphics accelerator architecture, data transfer to the GPU, textures and GPU buffers, GPU programming: shaders, basics of OpenGL, GLSL, CUDA and OpenCL. Advanced GPU techniques.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR004	Computer Graphics II	5	—	2/1 C+Ex
NPGR019	Realtime Graphics on GPU	5	—	2/1 C+Ex
NPGR010	Computer Graphics III	6	2/2 C+Ex	—
NPGR026	Predictive Image Synthesis Technologies	6	—	2/2 C+Ex
NPGR027	Shading Languages	5	—	2/1 C+Ex

b) Specialization ***Computer game development***

In addition to the two examination areas that are obligatory for all study branches, students get one question from each of the following examination areas:

1. Multi-agent systems (this area is obligatory for the specialization Computer game development).
2. Either the area Computer graphics for games or any area from the specialization Computer graphics.
3. One of the following areas: Parallel programming, System aspects of computers, Modern concepts of programming, Software engineering, Advanced programming, Web technologies.

In total, each student will get five questions.

Examination areas

1. Multi-agent systems (obligatory for the specialization Computer game development)
2. Computer graphics for games
3. Parallel programming
4. System aspects of computers
5. Modern programming concepts

6. Software analysis and architectures
7. Advanced programming
8. Web technologies

Knowledge requirements

1. Multi-agent systems

Autonomous agent architectures; agent perception, agent action selection mechanism, agent memory. Psychological inspiration. Methods for agent control; symbolic and connectionist reactive planning, hybrid approaches. Path search problem, steering rules, terrain representation. Communication and knowledge in multiagent systems, ontologies, speech acts, FIPA-ACL, protocols. Distributed problem solving, cooperation, Nash equilibria, Pareto efficiency, source allocation, auctions. Agent design methodologies, agent languages and environments. Ethological inspiration, models of population dynamics. Methods for agent learning; reinforcement learning, basic forms of animal learning.

Recommended courses

Code	Subject	Credits	Winter	Summer
NAIL054	Adaptive Agents	3	—	0/2 C
NAIL068	Human-like Artificial Agents	6	—	2/2 C+Ex
NAIL106	Multiagent Systems	6	—	2/2 C+Ex

2. Computer graphics for games

Homogeneous coordinates, affine and projective transformations in the plane and in space, quaternions, splines, interpolation by cubic splines, Bezier curves, Catmull-Rom splines, B-splines, de Casteljau and de Boor algorithms. Character animation, skinning, rigging. Collision detection.

2D Fourier transform and convolution, sampling and quantization, image anti-aliasing, texture, change of contrast and brightness, alpha-bending and compositing, compression of raster 2D graphics, video signal compression, temporal prediction (motion compensation), JPEG and MPEG standards.

Representation of 3D scenes, visibility, cast shadows, soft shadows, subsurface scattering, lighting models and shading algorithms, recursive ray tracing, physically-based model of light transport (radiometry, imaging equation), path tracing, precomputed global illumination, real-time global illumination, spherical harmonics-based shading, precomputed radiance transfer.

Graphics accelerator architecture, data transfer to the GPU, textures and GPU buffers, GPU programming: shaders, basics of OpenGL, GLSL, CUDA and OpenCL. Advanced GPU techniques. Game engine architecture.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPGR003	Introduction to Computer Graphics	5	2/2 C+Ex	—
NPGR004	Computer Graphics II	5	—	2/1 C+Ex
NPGR019	Realtime Graphics on GPU	5	—	2/1 C+Ex
NPGR033	Computer Graphics for Game Development	6	—	2/2 C+Ex

3. Parallel programming

Modern memory architectures (multiprocessor, GPU, NUMA), relationship between data location and access efficiency. Virtualization support. Converting computations for data parallelism, parallel solutions of non-homogeneous tasks. Efficient parallel implementation of basic algorithms. Synchronization, load balancing.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG042	Programming in Parallel Environment	6	—	2/2 C+Ex
NPRG054	High Performance Software Development	5	—	2/2 C+Ex

4. System aspects of computers

Process and thread management, communication (IPC), synchronization, memory models. Memory management (paging, heap structure, garbage collection). Runtime support, calling conventions (ABI), virtualization. Device management. File systems. Performance metrics, methods for performance evaluation, simulation and modeling. All topics considered in the context of contemporary parallel architectures.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI004	Operating Systems	5	2/1 C+Ex	—
NSWI131	Performance Evaluation of Computer Systems	5	—	2/1 C+Ex

5. Modern programming concepts

Modern object-oriented concepts, metaprogramming, reflection, aspects. Features of modern programming languages (for example annotations, iterators, generics, lambda functions). Code quality metrics, documentation, refactoring, testing and debugging. Design patterns. All topics considered in the context of contemporary programming languages.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex
NSWI068	Object and Component Systems	5	2/2 C+Ex	—

6. Software analysis and architectures

SW development processes, development phases. Business processes and their modeling using BPMN. UML and its use for analysis and design of structure and behavior of SW. Design patterns. SW testing, impact and change analysis. SW project planning, cost estimation, levels of project management. Legal aspects of SW, principal legal environment for IT projects. Types of SW architecture. Modeling and documentation of SW architecture. Classification of SW architecture quality attributes, their description using scenarios and tactics. Service oriented architectures. Algebraic methods, many sorted algebras, initial models. Formal principles of the UML language. OCL as a specification language, formal base of specification. Formal principles of RDF and OWL, the description logic.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI041	Introduction to Software Engineering	5	—	2/2 C+Ex
NTIN043	Formal Foundations of Software Engineering	5	2/2 C+Ex	—

7. Advanced programming

Object concepts of modern languages. Concepts of classless languages. Generic programming and metaprogramming, generics and templates, policies, traits, type inference. Modern constructs of programming languages. Object references and lifetime. Advanced aspects of imperative languages. Exceptions, exception-safe programming. Implementation of object properties, runtime support, calling conventions, garbage collection. Modern language constructs and code performance. Design patterns. Scripting languages, prototype-based languages. Domain-specific languages. Functional programming. Principles of code quality, best practices. Refactoring. Testing, debugging, monitoring.

Recommended courses

Code	Subject	Credits	Winter	Summer
NPRG059	Advanced Programming Praxis	2	0/1 C	—
NPRG014	Concepts of Modern Programming Languages	3	0/3 C	—
NPRG024	Design Patterns	3	—	0/2 MC
NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex
NSWI068	Object and Component Systems	5	2/2 C+Ex	—

8. Web technologies

Overview of basic web technologies. Network services for web technologies. Web services. Architecture of client-server applications, server-side and client-side scripting, web frameworks. Database systems in web applications, NoSQL databases, multimedia databases. Indexing and document searching, principles of web search engines. Linked Data, integration of semantic data to web pages. Security of information systems in the Internet environment, authentication, authorization, security models, cryptography basics, data security.

Recommended courses

Code	Subject	Credits	Winter	Summer
NSWI130	Software System Architectures	5	2/2 C+Ex	—
NSWI153	Advanced Technologies for Web Applications	5	—	2/2 C+Ex
NSWI145	Web Services	5	—	2/2 C+Ex
NDBI034	Multimedia Retrieval	4	2/1 C+Ex	—

NPRG043	Recommended Programming Practices	6	—	2/2 C+Ex