Syllabus for the course
"Theoretical computer science"

for doctoral programme in 09.06.01 Informatics and Computer Engineering / 05.13.01 "Systems Analysis, Control Theory, and Information Processing", 05.13.18 "Mathematical Modeling, Numerical Methods, and Software Systems"

Authors:
Bruno Bauwens, Assistant Professor, BrBauwens@gmail.com

Approved by the Academic Council of the Doctoral school in Computer Science on October 19, 2015

Moscow - 2015

1. Scope of Use

This program establishes the minimal requirements to doctoral students' knowledge and skills for 09.06.01 Informatics and Computer Engineering / 05.13.01 "Systems Analysis, Control Theory, and Information Processing", "05.13.18 Mathematical Modeling, Numerical Methods, and Software Systems" and determines the content of the course and educational techniques used in teaching the course.

The present syllabus is aimed at faculty teaching the course and doctoral students studying 09.06.01 Informatics and Computer Engineering / 05.13.01 "Systems Analysis, Control Theory, and Information Processing", 05.13.18 "Mathematical Modeling, Numerical Methods, and Software Systems".

This syllabus meets the standards required by:

• Educational standard of National Research University "Higher School of Economics" for 09.06.01 Informatics and Computer Engineering;

• Doctoral programme for 09.06.01 Informatics and Computer Engineering;

• University curriculum of the doctoral programme for 09.06.01 Informatics and Computer Engineering / 05.13.01 "Systems Analysis, Control Theory, and Information Processing", 05.13.18 Mathematical Modeling, Numerical Methods, and Software Systems".

2. Learning Objectives

The learning objective of the course "Theoretical Computer Science" is to provide doctoral students with theoretical background for their research in computer science.

- Automata and languages
- Computability theory
- Computational complexity
- Learning theory
- Cryptography
- Algorithmic game theory

3. Main Competencies Developed after Completing the Study of This Discipline

After completing the study of the discipline the doctoral student should:

- Recognize Turing complete problems as well as completes for NP, Pspace, etc.
- Understand the theory behind cryptography
- Understand basic limitations of algorithmic learning

Skills for research are developed, the students will learn to:

- Search and read abstract research papers
- Solve mathematically challenging problems
- Carefully write and present theoretical results.

After completing the study of the discipline the PhD student should have developed the following competencies:

Competence	Code	Descriptors (indicators of achievement of the result)	Educative forms and methods aimed at generation and development of the competence
The ability to carry out research in the field of professional activity using current research methods and information and communication technologies.	ОПК-1	Students obtain necessary knowledge to understand and formulate the theoretical difficulty of problems they are solving. Moreover, both for exercise lessons, the project and during the exam the students will be stimulated to search and share online information.	The student is allowed to use any information he find on the web for his project and during the exam.

The ability to carry out theoretical analysis and design of programming languages and systems, to use methods for analyzing program semantics.	ПК-2	Expressive power of several theoretical computing devices (Automata, Turing machines) and even a few programming languages will be investigated (topics 1, 2, and 3).	Examples covered during the lectures and exercises sessions.
The ability to develop and use methods for improving the efficiency and reliability of data and knowledge processing and transmission in computing machinery, systems, and networks.	ПК-3	A separate part (topic 4) is devoted to the mathematical foundations of cryptography.	Lectures and exercise sessions
The ability to do research in transformation of information into data and knowledge, models of data and knowledge representation, methods for knowledge processing, machine learning and knowledge discovery methods, principles of building and operating software for automation of these processes.	ПК-4	A separate part (topic 5) is devoted to learning theory. In this part we study the process of extracting models from data.	Lectures and exercise sessions

4. Place of the Discipline in the Postgraduate Program Structure

This is an elective course for 09.06.01 Informatics and Computer Engineering / 05.13.01 "Systems Analysis, Control Theory, and Information Processing", 05.13.18 "Mathematical Modeling, Numerical Methods, and Software Systems".

The following knowledge and competences are needed to study the discipline:

- Basic English language, both oral and written.
- Linear algebra.

5. Requirements and Grading

Project	1	A project that involves reading a chosen paper, carefully writing out technical details, and presenting the result in the classroom.
Exam	1	Written exam with discussion. Preparation time – 240 min.

6. Assessment

The assessment consists of a small project with a paper writing assignment and oral presentation.
Final assessment is the final exam. Students have to demonstrate knowledge of the material covered during the entire course and the ability to apply the materials.

7. The grade formula

The final exam is worth 60% of the final mark.
Final course mark *is obtained from the following formula*: Final=0.4* (Project: presentation + paper) + 0.6*(Exam).

All grades having a fractional part greater than 0.5 are rounded up.

Table of Grade Accordance

Ten-point Grading Scale	Five-point Grading Scale	
1 - very bad 2 – bad 3 – no pass	Unsatisfactory - 2	**FAIL**
4 – pass 5 – highly pass	Satisfactory – 3	**PASS**
6 – good 7 – very good	Good – 4	
8 – almost excellent 9 – excellent 10 – perfect	Excellent – 5	

8. Course description.

Topic 1. Automata and languages

Finite automata, regular expressions, non-regular languages, context-free grammers, non-context-free languages.

Topic 2. Computability theory

Turing machines, Turing completeness, Turing reduction, arithmetical hierarchy, Kolmogorov complexity, algorithmic randomness.

Topic 3. Computational complexity

Complexity classes P, BBP, NP, polynomial hierarchy, PSpace, Levin-Cook theorem, weak PCP theorem.

Topic 4. Cryptography and pseudorandomness

Semantic security, Goldreich-Levin theorem, Public key encryption, Zero knowledge proofs.

Topic 5. Computational learning theory

Polynomial time PAC learning, computational difficulty of various learning tasks, algorithmic sufficient statistics.

Topic 6. Algorithmic game theory

Nash-equilibrium: inefficiency, learning strategies and computational complexity, (online) auctions, applications in economics.

Topic 7. Topics suggested by the students

For example: combinatorial optimization, foundations of reinforcement learning, algorithmic graph theory, communication complexity, quantum computation, algorithmic information theory, etc.

9. Educational technologies

The following educational technologies are used in the study process:

- notes provided by the lecturer
- discussion and analysis in the exercise classes
- discussion on a supervised forum
- weekly consultation time
- student can hand in exercises to be corrected by the lecturer.

10. Final exam questions

The final exam will consist of a selection of problems. Students are allowed to use textbooks, notes and restricted internet access: they may download from any source but may not communicate on forums etc. Each question will require to solve mathematical problems using materials presented during the lectures and some of the project presentations.

To be prepared for the final exam, students will be given a sufficient amount of exercises. They can ask for hints and feedback on solutions.

11. Reading and Materials

Literature:

1. M. Sipser. *Introduction to the Theory of Computation*. Cengage Learning, 2006-2013.
2. S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

Literature for self-study:

3. J. Katz, & Y. Lindell. *Introduction to modern cryptography*. CRC Press, 2014.

12. Equipment.

- Blackboard. Computer room for the exam.