Government of Russian Federation

Federal State Autonomous Educational Institution
of High Professional Education

National Research University «Higher School of Economics»

Faculty of Computer Science
School of Data Analysis and Artificial Intelligence

Syllabus for the course
«Additional chapters of Discrete Mathematics:
Combinatorics, Graphs and Boolean logic»

for Bachelor degree specialisation
010400.62 «Applied Mathematics and Informatics»

Author:
Ilya Makarov, senior lecturer,
iamakarov@hse.ru

APPROVED BY

Approved on the meeting of the School of Data Analysis and Artificial Intelligence Head of the School Sergei O. Kuznetsov	Programme Academic Supervisor of specialisastion 010400 «Applied Mathematics and Information Science» Anton S. Konushin
____________________	____________________
«___» __________ 2014 г.	«___» __________ 2014 г.
Recommended by Academic Council of the Programme «Applied Mathematics and Information Science»	«___» __________ 2014 г.
Manager of the School of Data Analysis and Artificial Intelligence Larisa I. Antropova	____________________

Moscow, 2014

The syllabus must not be used by other departments of the university and other educational institution without permission of the department of the syllabus author

Additional chapters of Discrete Mathematics: Combinatorics, Graphs and Boolean logic.

Course Syllabus

I. Introduction: Subject and background

Tutor and author

Ilya A. Makarov, National Research University Higher School of Economics, Department of Data Analysis and Artificial Intelligence, Deputy Head

Teaching assistant

A. Munkoeva (possibly)

Summary

Additional Chapters of Discrete Mathematics: Combinatorics, Graphs and Boolean logic **(**ACDM:CGBL**) class covers more complicated sections of discrete mathematics.**

The combinatorics chapter is devoted to in-depth combinatorics, recursive sequences and the group action on the sets. The graph chapter covers theoretical foundations of algebric graph theory, algorithms on graphs and their applications. Boolean logic chapter covers modern methods for closed classes of Boolean logic, propositional calculus, predicate logic, and properties of classes to have a system of identities.

Cryptography applications is not included for the purpose to not interfere with parallel courses.

As a result, the ACDM:CGBL greatly advances students’ knowledge in the fields of modern discrete mathematics theory and applications, preparing our students to professional work in research projects and IT-industry.

Prerequisites

Basic C/C++ programming skills, basic notions from the first course of “Discrete mathematics” are required. No special skills are required, but all students are advised to have good fundamental mathematical level.

This course may be useful for collaboration with the following disciplines (names are given in Russian from official list of courses, more precise information can be viewed in synopsis):

Теория информации и кодирования
Упорядоченные множества и решётки
Прикладная теория графов
Компьютерная алгебра
Модели представления знаний и экспертные системы

Despite its theoretical content, the class has got many applications in the theory of algorithms, borrowing many math/IT concepts.

Aims

- To fill the gaps in modern problems of Discrete mathematics
- To learn practical problem-solving skills, which can be later applied in algorithmic theory
- To develop fundamental knowledge of combinatorics and complexity
- To develop practical skills needed in modern logic.
- To give practical knowledge, which is needed in many courses theoretical informatics

Background and outline

Some textbooks on discrete mathematics are written primarily for mathematics course, but the present book is intended for computing science students. The place of discrete mathematics in the undergraduate curriculum is now fairly well established, and it is certain that its place in the curriculum will be maintained in the third millennium.

Discrete mathematics has several aspects. One fundamental part is **combinatorics**, the study of counting arrangements of various types. We develop methods of counting which can deal with such problems. We begin from simple combinatorics and tend to recurrent sequences and action of a group on finite sets as a final resulting technique.

Next, **graph theory** can be used to model a variety of situations - road systems, chemical molecules, social networks, media interaction, etc. We introduce the basic types of graph and give some indication of what the important properties are that a graph might possess. We also describe boundaries for complexity of some basic algorithms.

At last, we deal with classic results in **Boolean logic** and its applications to many branches of mathematics. Such notions as completeness, finite generation, axiomatic completeness, and normal forms are considered. We also present a lot of counterexamples in many-valued logic and state open problems in this field.

Each chapter ends with a good number of examples and final programming task. The examples are a mixture of fairly straightforward applications of the ideas of the chapter and more challenging problems which are of interest in them or are of use later on in the book.

This class topic is new to HSE and Russian universities in general – it has some common points with the course of “Discrete mathematics”, which is read at department of Mechanics and Mathematics of Lomonosov Moscow State University, fourth course of specialist program. Modern tasks need strong perception and interpretation of problem on a pure mathematical language, clear understanding what we can do and what we can’t, and what are the conditions to optimize constructed algorithms and satisfy input and output limitations.

Teaching notes

The class is offered offline, with class material being 60% lectures and 40% classes.

Therefore, full student interaction with the tutor becomes the key to the class’ success. The students should not be taking this course in theory-only mode, but have to verify their skills in programming real tasks.

We use a combination of teaching tools and methodology.

- Good theory representation with complete interactive experience.
- Constant interaction with the students.
- Class projects. While homeworks are meant to demonstrate the understanding of the current class material, we use major programming tasks for individuals and groups of students.
- Rich educational software support for all chapters.
- “Office hours”. They are usually scheduled on request, for one or many students, based on their current needs and questions.

Teaching outcomes

The main outcome of this class is to give a student complete fundamental level of knowledge in combinatorics, graphs theory and Boolean logic. Our goal is to supplement theoretical knowledge with practical skills of program realizations and optimization.

Recommendations to the students

This class is meant to be interesting for further researches during studying on Theoretical informatics, Data Analysis and Artificial Intelligence specializations, has got upper-intermediate complexity. To anyone thinking about taking this class I would suggest the following:

- Take it only if you are interested in learning something new, and if you are not frightened of new math+programming class
- Be prepared to work both in classes and at home as well
- Do not miss/skip classes and homework.

II. Schedule

No	Topic	Total hours	In class hours		Self-study
			Lectures	Classes	
1	Sets, relations, functions, simplest combinatorial formulas	4	2	2	16
2	Euler formula for intersecting sets, Newton binomial, asymptotic combinatorial identities	4	2	2	8
3	Linear recurrent sequences and regular generating functions	6	4	2	12
4	Group action on finite sets	6	4	2	10
5	Graphs and trees, basic theorems on graphs and coloring of graph	10	6	4	12
6	Boolean logic and completeness of functional systems	8	6	2	12
7	Predicate logic, basic notions, prefix form	6	4	2	8
8	Normal forms, complexity for Boolean functions realizations by schemes and formulas	8	6	2	12
9	Basis and finite total equivalence systems for closed classes of Boolean logic	8	6	2	12
	Total: 162	**58**	**30**	**28**	**104**

III. Assessment

The assessment includes three main components:

- Class homework, assigned after each lecture
- Programs each month
- Final exam

The course is to be read in 3-4 modules.

Control type	Control form	Parameters
Current	Class work #1 (last week of third module) Class work #2 (last week of fourth module)	Written work during 120 minutes
Final	Exam	Speaking exam with preparation time about 60 minutes

Exam mark is calculated by formula:

$$O_{ex} = 0{,}2{*}O_{question\ 1} + 0{,}2{*}O_{question\ 2} + 0{,}3{*}O_{task} + 0{,}3{*}O_{add.\ q.}\ ,$$

where $O_{add.\ q.}$ is an additional mark for exam paper task

***Final course mark** is obtained from the following formula*:

$$O_{final} = 0{,}2{*}O_{class\ work\ 1} + 0{,}2{*}O_{class\ work\ 2} + 0{,}1{*}O_{classes} + 0{,}5{*}O_{ex}\ .$$

Correspondence table #1 for different types of marks

1-10	**Pass / Fail**
0-3	Fail
4-10	Pass

Correspondence table #2 for different types of marks

1-10	**2-5**
0 – 3	Unsatisfactory - 2
4 – 5	Satisfactory – 3
6 –7	Good – 4
8 – 10	Excellent – 5

IV. Synopsis

The importance of discrete mathematics has increased significantly since last 50 years, although there are few full handbooks of modern methods in its areas. The purpose of the course **ACDM:CGBL** is to provide systematic review of certain fields of discrete mathematics for computer scientists, mathematicians, and others, such as students, physical and social scientists, who need information about discrete mathematics.

The scope of material includes the many areas generally considered to be parts of discrete mathematics, focusing on the information considered essential to its application in computer science and engineering. Some of the fundamental topic areas covered includes:

- combinatorial designs
- recurrence relations
- generating functions
- computational geometry
- graph theory
- enumeration trees
- abstract algebra
- logic and set theory
- data structures and algorithms
- discrete optimization

V. Reading

Recommended:

1. *R. Garnier, J. Taylor*. Discrete Mathematics for New Technology, Second Edition, IOP, 2002
2. Handbook of discrete and combinatorial mathematics / *Kenneth H. Rosen*, editor in chief, *John G. Michaels*, project editor, CRC Press, 2000.
3. *Kenneth P. Bogart, Scot Drysdale, and Cliff Stein*, Discrete Math for Computer Science Students, 2004.
4. *O.B. Lupanov, A.B. Ugolnikov*. Selected papers.

In each part of a course three journal papers will be provided as a required reading on the current class subject.

Supplementary:

1. *Chen W.W.L.* Discrete mathematics, 1982.
2. *Andersen I.* A first course in discrete mathematics, Springer, 2002.
3. *L. Lov´ asz, K. Vesztergombi*. Discrete Mathematics Lecture Notes, Yale University, Spring 1999

VI. Final exam questions

The final exam will consist of two questions and one task, with 60-minute time limit to prepare an answer and solve a task. All questions will cover all three chapters, spanning the entire scope of the course.

VII. Topics for project and course work

- Many-valued logics “anti”-theorems and examples.
- Predicate approach in many-valued logics
- Labyrinth games

VIII. Typical class projects

- Group action on finite sets
- Algorithms on graphs and trees, problem of isomorphism
- Many-valued logics in comparison with Boolean logic

The syllabus is prepared by Ilya A.
Makarov