COURSE DESCRIPTION CARD - SYLLABUS

Course name
Discrete Mathematics

Course

Field of study
Arificial Intelligence

Year/Semester
1/1

Area of study (specialization)

Profile of study
general academic

Level of study
First-cycle studies

Course offered in
English

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes

Other (e.g. online)

Tutorials
30

Projects/seminars

Number of credit points

5

Lecturers

Responsible for the course/lecturer:
Maciej Machowiak, Ph.D.

Responsible for the course/lecturer:

Prerequisites

Students starting this course should have a basic knowledge of mathematics and computer science adequate to admission requirements. In addition, in the field of social competence, the student must present such attitudes as: honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

The course presents fundamentals - basic ideas and methods - of discrete mathematics, particularly from the field of logic, set theory, graph theory, transversal theory and combinatorics. Its general objective is to develop the student's capabilities for expressing ideas appearing in the field of computing in terms of functions and relations, applying logic principles, various proving techniques, recursion and approaches known from graph theory for solving problems related to computer science. Moreover, the course develops the students' ability of logical reasoning and expressing their concepts and ideas in an understandable way.

Course-related learning outcomes

Knowledge

Upon completion of the course the student:

1. has the general knowledge of discrete mathematics sufficient for formulating and solving complex problems arising in the field of computing using formal logic and set theory; knows fundamental laws of logic and set theory, as well as properties of relations and functions allowing for detecting and analysing relations and dependencies arising in computing problems;

2. has the knowledge necessary for formulating computing problems in terms of graph theory and allowing solving them with approaches known in this field;

3. has the knowledge necessary for formulating computing problems using various combinatorial objects and is aware of the relation between the number of these objects and the number of potential solutions of problems; knows counting principles allowing for determining the number of combinatorial objects and is aware of their importance for estimating time complexity of algorithms;

4. knows the principle of mathematical induction and is able to use inductive reasoning and recursion for formulating and solving computing problems;

5. knows the basic rules of determining the limiting behaviour of functions, necessary for estimating computational complexity of algorithms.

Skills

Upon completion of the course the student:

1. can use terms of discrete mathematics for formulating problems arising in the field of computing;

2. is able to apply methods of formal logic, set theory and graph theory for formulating and solving computing problems;

3. can use Latin squares and root polynomials for formulating and solving computing problems, particularly assignment problems;

4. is able to use methods of determining the limiting behaviour of functions - particularly asymptotic notations - for determining computational complexity of algorithms.

Social competences

Upon completion of the course the student:

1. is able to express herself/himself precisely and logically in terms of discrete mathematics.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes gained during lectures are verified based on the results of 90-minute written exam consisting of two tests: single-choice test and multiple-choice test. Each test contains close-ended questions related to the programme content presented by particular lecturers and it is evaluated separately. Positive grade is obtained by acquiring at least 50% of the maximum number of points in each test. Thresholds for particular grades are increased with 10%. The final grade is determined as a mean grade from both test results. Using auxiliary materials during the exam is not allowed.

Learning outcomes gained during tutorials are verified based on the results of two written tests consisting of a few open-ended questions (exercises). The final grade is determined based on the number of points gained during both tests. Positive grade is obtained by acquiring at least 50% of the maximum number of points. Using auxiliary materials during the tests is not allowed.

Programme content

Lectures cover the following topics:

1. Fundamentals of logic and set theory.

2. Relations and functions. Asymptotic notation.

3. Combinatorics: principles of counting, generating combinatorial objects, the rules of sum and product, variations with/without repetitions, permutations with/without repetitions, combinations with/without repetitions, set partition, binomial coefficient, multinomial coefficient.

4. Mathematical induction: the well-ordering principle, the first and second principle of mathematical induction.

5. Recursion: recurrence definitions, relations, problems and algorithms, mathematical induction for recursively defined sets.

6. Special numbers: Stirling numbers of the first and second kind, Bell numbers, Eulerian numbers of the first and second kind, harmonic numbers, Fibonacci numbers, Marsenne numbers.

7. Introduction to graph theory: directed and undirected graphs, paths, walks, cycles, Eulerian and Hamiltonian graphs, graph colouring, trees, etc.

8. Properties of integers: primes, divisibility rules, etc.

9. Fundamentals of transversal theory: Hall's theorem, tournaments, minimax theorems.

10. Advanced principles of counting: inclusion and exclusion principle, Dirichlet's principle - the pigeonhole principle.

12. Latin squares: Latin square design, orthogonal Latin squares.

13. Rook polynomials: decomposition.

Teaching methods

1. Lectures: multimedia presentations of programme contents with numerous examples.

2. Tutorials illustrating programme contents presented during lectures with exercises solved by teachers using a blackboard or by students eager to present their ideas. Discussions in a student group on the ideas proposed by particular students.

Bibliography

Basic

1. Discrete and combinatorial mathematics. An applied introduction, R.P. Grimaldi, Addison Wesley Publishing Company, New York, 1999.

2. Discrete Mathematics, Kenneth A. Ross, Charles R.B. Wright, Prentice Hall, 1992

3. Concrete Mathematics, Donald E. Knuth, Ronald L. Graham, Oren Patashnik, Addison-Wesley, 1994

4. Matematyka dyskretna, K.A. Ross, Ch.R.B. Wright, PWN, Warszawa, 2012.

5. Matematyka konkretna, R.L. Graham, D.E. Knuth, O. Patashnik, PWN, Warszawa, 2012.

Additional

1. Discrete Mathematics and its Applications, Kenneth H.Rosen, McGraw-Hill, 2007

2. Discrete Mathematics, Swapan K. Chakraborty, Oxford University Press, 2011

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,0
Classes requiring direct contact with the teacher	60	2,5
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation) [1]	65	2,5

[1] delete or add other activities as appropriate