MODULE DESCRIPTION

Module code	
Module name	**Matematyka Dyskretna**
Module name in English	**Discrete Mathematics**
Valid from academic year	**2012/2013**

MODULE PLACEMENT IN THE SYLLABUS

Subject	**Computer Science**
Level of education	**1st degree** *(1st degree / 2nd degree)*
Studies profile	**General** *(general / practical)*
Form and method of conducting classes	**Full-time** *(full-time / part-time)*
Specialisation	
Unit conducting the module	**The Department of Computer Science Applications**
Module co-ordinator	**Radosław Pytlak, PhD hab., Eng., Professor of the University**
Approved by:	

MODULE OVERVIEW

Type of subject/group of subjects	**Major** *(basic / major / specialist subject / conjoint / other HES)*
Module status	**Compulsory** *(compulsory / non-compulsory)*
Language of conducting classes	**Polish**
Module placement in the syllabus - semester	**3rd semester**
Subject realisation in the academic year	**Winter semester** *(winter / summer)*
Initial requirements	**Mathematical Analysis and Algebra** *(module codes / module names)*
Examination	**Yes** *(yes / no)*
Number of ECTS credit points	**6**

Method of conducting classes	Lecture	Classes	Laboratory	Project	Other
Per semester	**30**	**30**			

TEACHING RESULTS AND THE METHODS OF ASSESSING TEACHING RESULTS

Module target	The aim of the module is to familiarise students with the elements of mathematics necessary in formulating tasks from the field of computer science in the mathematical language as well as in solving the formulated mathematical tasks.

Effect symbol	Teaching results	Teaching methods *(l/c/l/p/other)*	Reference to subject effects	Reference to effects of a field of study
W_01	A student has basic knowledge as regards the set theory, mathematical logic, the relation theory and the graph theory necessary in formulating and solving computer science tasks.	l/c	K_W03	T1A_W01,T 1A_W02 InzA_W02
W_02	A student has systematised knowledge as regards the methods of solving tasks concerning the set theory, the relation theory, and the graph theory connected with applying mathematics in computer science.	l/c	K_W03 K_W07	T1A_W01,T 1A_W02 T1A_W03 InzA_W02
U_01	A student is able to utilise the acquired knowledge to formulate tasks from the field of computer science in the mathematical language.	l/c	K_U09	T1A_U09 InzA_U02
U_02	A student can utilise the learnt methods of solving tasks from the field of the set theory, mathematical logic, the relation theory, and the graph theory in solving computer science problems.	l/c	K_U10	T1A_U10,T1 A_U13 InzA_U05
U_03	A student is able to utilise the learnt models and methods in independent problem description of computer science problems in the mathematical language.	l/c	K_U10	T1A_U10,T1 A_U13 InzA_U05
K_01	A student understands the necessity of continuous self-education and learns the examples and various aspects of utilising the set theory, mathematical logic, the relation theory, and the graph theory in computer science.	l/c	K_K01, K_K02	T1A_K01, T1A_K02, InzA_K01

Teaching contents:

Teaching contents as regards lectures

Lecture number	Teaching contents	Reference to teaching results for a module
1	**The elements of the set theory** The concept of a set and its elements. The numerical size of the set. Power set. Basic laws of the set theory.	W_01, W_02
2	**Functions and relations** The definition of a function, domain, and codomain. The classification of a function. The elements of a functional analysis. The definition of a relation. The classification of ordering relations. Equivalence relations. Spaces with relations. The concept of least and greatest elements; the concept of greatest lower bound and least upper bound. The Hasse diagram.	W_01 W_02 K_01
3	**The elements of mathematical logic**	W_01

	Propositional logic. Logically equivalent sentences. Logical implications. The methods of proving statements. The methods of obtaining logically equivalent sentences.	
4	**Mathematical induction** Four principles of mathematical induction. The notion of an algorithm loop. Loop invariants theorem.	W_01 W_02 K_01
5	**Recurrent dependencies** Homogenous and non-homogenous recurrence relations. Solving second-order homogenous recurrence relations. Solving the selected recurrence relations. The Euclidean algorithm.	W_01 W_02
6	**The elements of combinatorics** The basics of summing techniques. Basic probability calculus.	W_01
7	**Introduction to the graph theory** The notion of a graph as a three-fold relation. The methods of defining a graph – adjacency matrices and graph incidence. The characteristics of graph apexes and branches. The notion of a graph and subgraph part/element.	W_01 W_02
8	**The selected algorithms of the graph theory** Utilising Boolean function to determine graph minimal bases. Utilising Boolean functions to determine the set minimum covers of graph apexes. The colouring algorithm of graph apexes. The algorithm of determining the shortest chains in a graph. The algorithm of determining the components of graph cohesion. The algorithm of determining Euler's chain.	W_01 W_02 K_01

Teaching contents as regards classes

Class number	Teaching contents	Reference to teaching results for a module
1	**Functions and relations** The definition of a function, domain, and codomain. The classification of a function. The elements of a functional analysis. The definition of a relation. The classification of ordering relations. Equivalence relations. Spaces with relations. The concept of least and greatest elements; the concept of greatest lower bound and least upper bound. The Hasse diagram.	W_01, W_02 U_01 U_02
2	**The elements of mathematical logic and mathematical induction** Propositional calculus. Logically equivalent sentences. Logical implications. The methods of proving statements. The methods of obtaining logically equivalent sentences. The application of mathematical induction principles in proving statements/theorems on the set of natural numbers.	W_01, W_02 U_01
3	**Recurrence relations** Homogenous and non-homogenous recurrence relations. Solving second-order homogenous recurrence relations. Solving the selected non-homogenous recurrence relations.	W_01, W_02 U_01 U_02
4	**Introduction to the graph theory** The notion of a graph as a three-fold relation. The methods of defining a graph - adjacency matrices and graph incidence. The characteristics of graph apexes and branches. The notion of a graph and subgraph element.	W_01, W_02 U_01

5	**The selected algorithms of the graph theory** Utilising Boolean function to determine graph minimal bases. Utilising Boolean functions to determine the set minimum covers of graph apexes. The colouring algorithm of graph apexes. The algorithm of determining the shortest chains in a graph. The algorithm of determining the components of graph cohesion. The algorithm of determining Euler's chain.	W_01, W_02 U_01 U_02

The methods of assessing teaching results

Effect symbol	**Methods of assessing teaching results** *(assessment method, including skills – reference to a particular project, laboratory assignments, etc.)*
W_01	A test on the principles of mathematical induction, mathematical logic, and the relations theory.
W_02	A test on the elements of the graph theory.
U_01	A test on the ability of solving tasks concerning mathematical induction, proving methods, and the relation theory.
U_02	A test on the ability of utilising algorithm in solving tasks concerning the graph theory.
U_03	A test on the ability of utilising algorithm in solving tasks concerning the graph theory.
K_01	A test on the ability of utilising algorithm in solving tasks concerning the graph theory.

STUDENT'S INPUT

ECTS credit points		
	Type of student's activity	**Student's workload**
1	Participation in lectures	**30**
2	Participation in classes	**30**
3	Participation in laboratories	
4	Participation in tutorials (2-3 times per semester)	**2**
5	Participation in project classes	
6	Project tutorials	
7	Participation in an examination	
8		
9	**Number of hours requiring a lecturer's assistance**	**62** *(sum)*
10	**Number of ECTS credit points which are allocated for assisted work** *(1 ECTS credit point=25-30 hours)*	**2.37**
11	Unassisted study of lecture subjects	**40**
12	Unassisted preparation for classes	**15**
13	Unassisted preparation for tests	**40**
14	Unassisted preparation for laboratories	
15	Preparing reports	
16	Preparing for a final laboratory test	
17	Preparing a project or documentation	
18	Preparing for an examination	
19	Preparing questionnaires	
20	**Number of hours of a student's unassisted work**	**95** *(sum)*

21	**Number of ECTS credit points which a student receives for unassisted work** *(1 ECTS credit point=25-30 hours)*	**3.63**
22	**Total number of hours of a student's work**	**157**
23	**ECTS credit points per module** *1 ECTS credit point=25-30 hours*	**6**
24	**Work input connected with practical classes** *Total number of hours connected with practical classes*	**105**
25	**Number of ECTS credit points which a student receives for practical classes** *(1 ECTS credit point=25-30 hours)*	**4.01**