1.3. Module/ course form

To be completed by Course Team

Module name : **Algorithms and Data Structures**		Module code:
Course name: **Algorithms and Data Structures**		Course code:
Faculty: **The Institute of Applied Informatics**		
Field of study: **Informatics**		
Mode of study : **Full-time**	Learning profile: **PRACTICAL**	Speciality:
Year/ semester:	Module/ course status: **mandatory**	Module/ course language: **polish/english**

Type of classes	lecture	lessons	lab	project	tutorial	other (please specify)
Course load	**30**		**30**			

Module/ course coordinator	dr inż. Henryk Olszewski
Lecturer	dr inż. Henryk Olszewski, mgr inż. Daria Rybarczyk, mgr inż. Mariusz Mazurek
Module/ course objectives	The purpose of the lecture is to present the theoretical construction of algorithms, and information on their use, which are verified during laboratory classes. Problems associated with the analysis of accuracy and complexity of algorithms and data structures, operations performed on them, and examples of their use will be presented. Students will gain ability to construct, write and analyze algorithms.
Entry requirements	Knowledge of mathematics in the secondary school. The basic principles of the use of information technology.

LEARNING OUTCOME		
Nr	LEARNING OUTCOME DESCRIPTION	Learning outcome reference
1	Student has a general knowledge of the methods and principles of algorithms creating and analysis.	K_W07
2	Student has a general knowledge of the basic and complex data structures.	K_W07
3	Student has a detailed knowledge of the writing algorithms in a programming language.	K_W07
4	Student has the ability to build algorithms, such as calculation and sorting algorithms.	K_U17
5	Student is able to estimate the computational and memory complexity of algorithms.	K_U17
6	Student has the ability to use algorithms available in the programs.	K_U17
7	Student is able to think creatively.	K_K05

8	Student understands the need and knows the possibility of continuous learning.	K_K01
9	Student is responsible for his own work and the work of the team.	K_K04

CURRICULUM CONTENTS

Lecture

- The computer as a tool for solving of specific tasks. Diagram of solving of the problem - the algorithm. Definitions of algorithm. (2 hrs.)
- Flowcharts of algorithms. Components of flowcharts. (2 hrs.)
- Software for algorithms creating – ELI program. (2 hrs.)
- Algorithms with branching. Iterations and algorithms that use iterations. (2 hrs.)
- Recurrent algorithms. Fibonacci series. Dangers of recursion – unreality code cases, elite cases. Derecursivation. (2 hrs.)
- The theory of fractals. Sierpinski triangle. Pascal's triangle. Reduction factor. Mandelbrot set. Julia set. Fractal face of Newton's method. Lorenz attractor. (2 hrs.)
- Sorting algorithms. Bubble sort. Sort by choice. Bucket sort and sort file by position. (2 hrs.)
- The method of "divide and conquer". Merge sort. (2 hrs.)
- The computational complexity of the algorithm. The medium and pessimistic complexity. (2 hrs.)
- QuickSort algorithm. (2 hrs.)
- Properties of sorting algorithms. Sorting in place. The complexity and efficiency of sorting algorithms. (2 hrs.)
- Elementary structure of an data. Stack. Queue. List. Tree. Heap. Biparental heap (2 hrs.)
- Elements of graph algorithms. The data structure used to represent the graph. The methods of the graph viewing. (2 hrs.)
- Create models of phenomena. The use of algorithms in mathematics, physics, astronomy, and economics. Measuring and control algorithms. (2 hrs.)
- Completion of the course. (2 hrs.)

Lab

The purpose of the laboratory is the teaching of construction and analysis of algorithms for solving specific tasks using the computers, ELI program, Pascal or C languages.

- Flowcharts of algorithms. Components of flowcharts. Software for algorithms creating – ELI program. (1 hr.)
- Algorithms with branching. Iterations and algorithms that use iterations. (4 hr.)
- Recurrent algorithms. Fibonacci series. Dangers of recursion – unreality code cases, elite cases. Derecursivation. (4 hrs.)
- The theory of fractals. Sierpinski triangle. Pascal's triangle. Reduction factor. Mandelbrot set. Julia set. Fractal face of Newton's method. Lorenz attractor. (1 hrs.)
- Sorting algorithms. Bubble sort. Sort by choice. Bucket sort and sort file by position. (3 hrs.)
- The method of "divide and conquer". Merge sort. (2 hrs.)
- The computational complexity of the algorithm. The medium and pessimistic complexity. (3 hrs.)
- QuickSort algorithm. (2 hrs.)
- Elementary structure of an data. Stack. Queue. List. Tree. Heap. Biparental heap (5 hrs.)
- Elements of graph algorithms. The data structure used to represent the graph. The methods of the graph viewing. (3 hrs.)
- Create models of phenomena. The use of algorithms in mathematics, physics, astronomy, and economics. Measuring and control algorithms. (1 hr.)
- Completion of the course. (1 hr.)

Basic literature	1. Banachowski L., Diks K., Rytter W.: Algorytmy i struktury danych. WNT, Warszawa, 1996. 2. Cormen T. H., Leiserson C. E., Rivest R. L.: Wprowadzenie do algorytmów. WNT, Warszawa, 1997. 3. Sysło M. M. Algorytmy. WSiP Warszawa, 2005. 4. Wróblewski P.: Algorytmy, struktury danych i techniki programowania. Wydawnictwo Helion, Gliwice 2001. 5. Wirth N.: Algorytmy + struktury danych = programy. WNT, Warszawa 1980. 6. Aho A.V., Hopcroft J.E., Ulman J.D.: Projektowanie i analiza algorytmów komputerowych. PWN, Warszawa 1983.
Additional literature	1. Janiak A.: Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów. Akademicka Oficyna Wydawnicza PLJ, 1999. 2. Wanat K.: Algorytmy numeryczne. Wydawnictwo Helion, Gliwice 1993. 3. Jaszkiewicz A.: Inżynieria oprogramowania, Wydawnictwo Helion, Gliwice 1997. 4. Sysło M. M.: Elementy Informatyki. Podręcznik dla nauczyciela. Wydawnictwo Naukowe PWN, Warszawa, 1997. 5. Sysło M. M.: Piramidy, szyszki i inne konstrukcje algorytmiczne. WSiP Warszawa, 1998.

Teaching methods	Lecture and multimedia presentation, laboratory exercises, discussion, problem solving, teamwork and individual work in the computer laboratory.

Assessment method	Learning outcome number
Execution of the tasks of constructing of selected algorithms using algorithmic techniques of and the data structures.	01, 02, 04
Implementation tasks of analyzing the complexity of the algorithm.	01, 02, 05
Execution of the tasks of writing the algorithm in a programming language.	03, 06, 07, 08, 09

Form and terms of an exam	Laboratory assessment: completion of all laboratory activities under the program, pass test, consisting of all the material covered in the laboratory exercises. If you do not pass test, you can pass it to the test correction. Terms of pass the course: completion of the laboratory, a positive result of an exam.

STUDENT WORKLOAD

	Number of hours
Participation in lectures	30
Independent study of lecture topics	10
Participation in tutorials, labs, projects and seminars	30
Independent preparation for tutorials*	60
Preparation of projects/essays/etc. *	
Preparation/ independent study for exams	15
Participation during consultation hours	5
Other	2

TOTAL student workload in hours	152
Number of ECTS credit per course unit	**6 ECTS**
Number of ECTS credit associated with practical classes	90 **3,6 ECTS**
Number of ECTS for classes that require direct participation of professors	67 **2,7 ECTS**