(faculty stamp)

COURSE DESCRIPTION

1. Course title: THEORY OF COMPUTER SCIENCE	**2. Course code: PI**
3. Validity of course description: 2018/2019	
4. Level of studies: 1st cycle of higher education	
5. Mode of studies: intramural studies	
6. Field of study: INFORMATICS (RAU)	
7. Profile of studies: academic	
8. Programme: ALL	
9. Semester: 1, 2	
10. Faculty teaching the course: Instytut Informatyki	
11. Course instructor: dr inż. Alina Momot	
12. Course classification: common subject	
13. Course status: compulsory	
14. Language of instruction: English	
15. Pre-requisite qualifications: none	

16. Course objectives:

The aim of the lecture is to delivery to students the information in the range of the basic notions of computer science. The aim of the classes and laboratory is to purchase by the students the skill in the range of creating the algorithms, low-level programming, understanding of works of microprocessors and introduction with the basic structures of the data.

The basic issues presented in the course are:
- the concept of the algorithm
- semaphores, classic synchronization problems (producer-consumer, problem of five philosophers)
- machine data representation and implementation of arithmetic operations
- organization of the central processor unit
- assembler language
- basic information about computer networks
- an overview of data access methods

17. Description of learning outcomes:[1]

Nr	Learning outcomes description	Method of assessment	Teaching methods	Learning outcomes reference code
1	A student knows formal methods of algorithms notation	Written test	Lecture, class, laboratory	K1A_W09

[1] należy wskazać ok. 5 – 8 efektów kształcenia

2	A student has theoretically founded knowledge in the field of formal grammars. He can simulate simple algorithms using the Turing machine	Written test	Lecture, class	K1A_U10
3	A student knows the basic construction elements of the computer and can discuss the role of each of them. He knows how the computer works according to the von Neumann model. He can explain the role of the computer control system, he can define simple instructions for the given lecture computer architecture (so-called W machine).	Written test, laboratory, exam	Lecture, class, laboratory	K1A_W05, K1A_W13, K_U23
4	A student can write simple programs in the assembler language (W machine)	Written test, laboratory, exam	Lecture, class, laboratory	K1A_U23
5	A student is able to define ways to protect indivisible resources and explain what is the management of resources and deadlock	Written test, laboratory, exam	Lecture, class, laboratory	K1A_U19
6	A student can use a queuing model to model a single computer and a computer network and analyse this model	Written test	Lecture, class	K1A_W11, K1A_U08, K1A_U12

18. Teaching modes and hours

	Lect.	C.	Lab.	P.	Sem.
Semester 1	**30**	**30**	**30**	**-**	**-**
Semester 2	**15**	**15**	**-**	**-**	**-**

19. Syllabus description:

Lecture:
Algorithms. Concept and examples of algorithms, notation, serial and parallel implementation of algorithms. Complexity, time and memory complexity and relationships between them, examples of algorithm complexity evaluation for serial and parallel sorting algorithms.
The operating system and its tasks. The role and functions of the operating system. Synchronization and communication between processes. The concept of system resources. Sharable and unsharable resources. TAS procedure, semaphores. Producer-consumer problem. Resource management issues, deadlock protection.
Computer design elements. Registers, counters, encoders and decoders. The arithmetic and logic unit, its design and implementation. Buses and problems related to their design. Memories: types, design and implementation.
Computer W design and implementation. Control signals and their role. Command cycle and computer command equations. Synthesis of a computer control system based on its command equations. Intelligent phase distributors. Microprogram memory, micro-instructions, distribution nodes.
Interrupts, the concept, organization of the interrupt system. Interrupt priorities.
Information exchange between the central unit and the environment. Concepts of elementary I/O system. The exchange circuit as an intermediary between the central unit and external devices. Construction of an exemplary exchange circuit. Direct memory access (DMA), construction and operation.
Addressing modes.
Programming languages. Machine code, assembler, assembly process, macro-assembler, examples of macro-definition, segmentation. Higher level languages, compilers. Examples of compilation of program fragments in a high-level language. Algorithm for compilation of an arithmetic expression.
Turing machine, description, Church-Turing thesis, a characteristic table, examples of programming algorithms for Turing machine.
Formal languages. Definition of grammar, examples of grammars, BNF notation, Łukasiewicz's algebra, the notion of translation and an example of translation algorithm of arithmetic expressions from infix notation to reverse Polish

notation.
Data access methods. Lists, binary trees, network structures and their analysis in terms of search time. B-tree structures and task with optimal structure. Hashing functions and database structures based on them, collision problem.
Parallel computers. Notation of the algorithm in a canonical form, theorem on the implementation of the algorithm, serial and parallel implementation, definition of acceleration, model of data-flow computer and examples of algorithm implementation in such systems
Computer networks and their statistical models. The concept of a computer network, media and their characteristics, network topologies, a layered model. Statistical computer model as a service station (M / M / 1) and its analysis. Computer network model as M / M / 1 network. Open and closed networks. Methods for calculating load distribution in the network.
The latest trends in information technology. Information technology and genetics, quantum information systems.

Class

The subject of classes is closely related to lectures and is an extension and illustration of selected problems presented during the lectures. Some of the tasks are solved by the students themselves, when solving them or after solving, there is often a discussion that allows to consider alternative ways of solving the problem. The topics of table exercises include the following issues:

- Algorithms.
- Recursion and examples of recursive algorithms.
- Semaphores and resource management.
- Design of computer instructions
- Assembler programming
- Turing machine
- Formal grammars
- Data access methods
- Queue system M/M1 and networks

Laboratory

Laboratory exercises are conducted on the first semester of full-time studies. During the laboratory, students solve the tasks indicated by the teacher, using specially prepared software for this purpose. The following exercises are carried out as part of the laboratory:

1. Recursive algorithms
2. Synchronization and communication between processes
3. Instructions design for the machine W
4. Programming in the assembly language of the W machine
5. Input / output system
6. Interrupt system

20. Examination: yes (after the first semester)

21. Primary sources:

1. Andrew S. Tanenbaum, Todd Austin: *Structured Computer Organization*. Pearson, 2013

2. John L. Hennessy, David A. Patterson: *Computer Architecture, A Quantitative Approach.* Morgan Kaufmann, 2012

22. Secondary sources:

1. William Stallings: *Computer Organization and Architecture, Designing for Performance*. Pearson, 2016
2. Noam Nisan, Shimon Schocken: *The Elements of Computing Systems. Building a Modern Computer from First Principles*. MIT Press, 2005
3. Allen B. Downey: *The Little Book of Semaphores,* 2016

23. Total workload required to achieve learning outcomes

Lp.	Teaching mode	Contact hours / Student workload hours
1	Lecture	45/45
2	Classes	45/45
3	Laboratory	30/30
4	Project	/
5	BA/MASeminar	/
6	Other	/
	Total number of hours	120/120

24. Total hours: 240

25. Number of ECTS credits:[2] 8 (respectively in sem. 1 and 2: 6 + 2)

26. Number of ECTS credits allocated for contact hours: 4

27. Number of ECTS credits allocated for in-practice hours (laboratory classes, projects): 2

26. Comments:

Approved:

……………………………. *(date, Instructor's signature)*

………………………………………………… (*date , the Director of the Faculty Unit signature)*

[2] 1 punkt ECTS – 30 godzin.