Subject (course) name: **Computer Architecture**		
Programme: **Computer Science** Specialty:		Subject code: **15**
		Title graduate: **Engineer**
Type of course: **obligatory**	Course level: **First-cycle studies**	Year: **II** Semester: **III** Semester: **spring**
Form of classes: **Lectures, Classes, Labs, Seminar, Project**	Number of hours per week: **2L, 0, 1Lab, 0, 0**	Credit points: **4 ECTS**

GUIDE TO SUBJECT

SUBJECT OBJECTIVES

C1. General knowledge in number representation, format conversion, computer arithmetic.

C2. Particular knowledge in computer architecture and organization.

C3. General ability to classify, compare and characterize the primary features of real computers.

C4. General programming skills in low-level language.

SUBJECT REQUIREMENTS

1. General knowledge in math and logical devices.
2. General ability to programming.
3. General ability to independently search in literature and online resources.

LERNING OUTCOMES

EK 1 - Student will be able to represent numbers in various formats, convert them and will collect knowledge in computer arithmetic operation.

EK 2 - Student will be able to characterize a computer and its elements.

EK 3 - Student will be able to classify the real computers on the basis of its parameters and documentation.

EK 4 - Student will be able to design a simple software in low level programming and report it.

SUBJECT CONTENT

Form of classes - lectures

Topic	Hours
W1 – Introduction – history of computers	**1**
W2 – Basic computer ideas: von Neumann and Harvard architectures	**2**
W3 – CPU: elements, instruction cycle, instruction set	**2**
W4 – Floating point representation, IEEE-P754 standard, FPU	**2**
W5 – Pipeline computing	**2**
W6 – Superscalar	**2**
W7 – Parallel processing, supercomputers, vector computing, multicore processors	**2**
W8 – Memory: hierarchy, association, virtual memory, cache coherency	**2**
W9 – Input-output circuits, interrupts and exceptions	**2**
W10 – Overview of modern computers and alternative architectures	**2**
W11 – Number representation, format conversion	**2**
W12 – Arithmetic operations in a fixed- and floating point formats	**7**

Final test	**2**
Total	**30**

Form of classes – laboratory

Topic	**Hours**
L1 – Driving microcontroller's port lines	**2**
L2 – Program and data memories, addressing modes	**2**
L3 – Arithmetic operations, stack, subroutines	**2**
L4 – Driving the 7-segment LED display	**2**
L5 – Reading the sequential keyboard	**2**
L6 – Reading the matrix keyboard	**2**
L7 – Driving the alphanumerical LCD display	**2**
Examination of programming tasks	**1**
Total	**15**

STUDY METHODS

1. Lectures using multimedia presentations and computer arithmetic tasks
2. Discussion during the course and in addition during individual consultations
3. Laboratory – analysis of the operation and development of software - teamwork

EDUCATIONAL TOOLS

1. Audiovisual equipment, black(white)board, lectures in electronic version
2. Textbook with exercises of computer arithmetic
3. Dedicated software for presentation of chosen aspects discussed during lectures
4. Microprocessor based development systems - labs

METHODS OF ASSESMENT (F – Forming, P – Summary)

F1. assessment of self preparation for laboratory classes – oral answer
F2. assessment of the correctness and timeliness of presentation software created
P1. lecture – written test of the theory and completion of tasks in computer arithmetic
P2. laboratory – assessment of ability to software analysis and software development

STUDENT WORKLOAD

Form of activity		Averaged workload (hours)		
		[h]	∑ [h]	ECTS
Participation in class activities	lecture laboratory consultation	30 15 3	48	3
Preparation for tutorials (reading literature)		10	42	1
Preparation for test		15		
Preparation for computer arithmetic tasks		15		
Familiarizing yourself with the educational software		2		
Total			**90**	**4**

A. BASIC READING

1. Patterson D., Hennessy J.: Computer Organisation and Design: The Hardware/Software Interface, Morgan Kaufmann, 2009.
2. Parhami B.: Computer Arithmetic: Algorithms and Hardware Designs, 2nd edition, Oxford University Press, New York, 2010.
3. Mano M.: Computer System Architecture, Pearson Education, 2008.
4. Stallings W.: Computer Organization and Architecture, Designing for performance, 8th edition, Pearson Education, 2008.

B. FURTHER READING

1. Baer J.L.: Microprocessor Architecture. From Simple pipelines to Chip Multiprocessors. Cambridge University Press, New York 2010.
2. Grys S.: Arytmetyka komputerów w praktyce (en. Computer Arithmetic in Practice). Wyd. Naukowe PWN, Warsaw 2007 (reprint 2013) – in Polish.
3. Journals, e.g..: IEEE Transactions on Computers, IEEE Computer, IEEE Parallel and Distributed

Technology, Computers & Electrical Engineering.

Learning objectives	In relation to the learning outcomes specified for the field of study	Subject objectives	Study methods	Methods of assessment
EK1	K_W07 K_U13	C1	lectures, discussion	P1
EK2	K_W07	C2	lectures, discussion	P1
EK3	K_W07	C3	lectures, discussion	P1
EK4	K_W05 K_U02 K_U13	C4	discussion, laboratory	F1,F2,P2

II. EVALUATION

Grade	Outcome
EK1	**Student is able to represent numbers in various formats, convert them and will collect knowledge in computer arithmetic operation**
2 (F)	Student is not able to represent numbers in various formats, convert them and did not collect knowledge in computer arithmetic operation
3 (E)	Student is able to represent numbers in various formats
4 (C)	Student is able to represent numbers in various formats and convert them
5 (A)	Student is able to represent numbers in various formats, convert them and will collect knowledge in computer arithmetic operation
EK2	**Student is able to characterize a computer and its elements**
2 (F)	Student is not able to characterize a computer and its elements
3 (E)	Student is able to characterize a computer
4 (C)	Student is able to characterize a computer and its some elements
5 (A)	Student is able to characterize a computer and its elements
EK3	**Student is able to classify the real computers on the basis of its parameters and documentation**
2 (F)	Student is not able to classify the real computers on the basis of its parameters and documentation
3 (E)	Student is able to classify the real computers on the basis of its primary parameters
4 (C)	Student is able to classify the real computers on the basis of its primary parameters
5 (A)	Student is able to classify the real computers on the basis of its parameters and documentation
EK4	**Student is able to design a simple software in low level programming and report it**
2 (F)	Student is not able to design a simple software in low level programming
3 (E)	Student is able to design a simple software in low level programming on the basis of demo software
4 (C)	Student is able to design a simple software in low level programming but has limited ability to report it
5 (A)	Student is able to design a simple software in low level programming and report it

III. OTHER USEFUL INFORMATION

1. All information for students on the schedule are available on the notice board and on the website: https://we.pcz.pl/
2. Information on the consultation shall be provided to students during the first lecture and will be placed on the website https://we.pcz.pl/
3. Terms and conditions of credit courses will be provided to students during the first lecture