MODULE DESCRIPTION

Module title	Module code
Computer architecture	

Lecturer(s)	Department where the module is delivered
Coordinator: assoc. prof. dr. Antanas Mitašiūnas **Other lecturers:**	Department of Computer Science Faculty of Mathematics and Informatics Vilnius University

Cycle	Type of the module
First	Compulsory

Mode of delivery	Semester or period when the module is delivered	Language of instruction
Face-to-face	1 semester	Lithuanian

Prerequisites
Prerequisites:

Number of credits allocated	Student‘s workload	Contact hours	Self-study hours
5	142	68	74

Purpose of the module: programme competences to be developed

Purpose of the module – to shape understanding of the real processing of computer programs as iterative transformation of memory data state using computer's instructions, to master the system of machine level notions, to learn read and write machine level software.

Generic competences:

- Communication and collaboration *(GK1).*
- Life-long learning *(GK2).*

Specific competences:

- Knowledge and skills of underlying conceptual basis *(SK4).*
- Technological and methodological knowledge and skills, professional competence *(SK6).*

Learning outcomes of the module: students will be able to	Teaching and learning methods	Assessment methods
• operate computer architecture concepts and notions fluently and focused;	Teaching methods: • Lectures; • Laboratory works. Learning methods: • Actual knowledge gathering and accumulation; • Knowledge synthesis – generalization, abstraction and aggregation of actual knowledge; • Knowledge analysis – new knowledge matching with aggregated knowledge, their verification and correction is needed; • Application of aggregated and validated knowledge.	Examination. Laboratory works presentation. Criteria: • Use of exercises to assess understanding and ability to apply theoretical knowledge • Demonstration of ability to develop, explain and modify software in assembler language
• understand computer systems diagnostic messages done in machine oriented terms;		
• develop programs in assembler language;		
• posses concepts needed to learn programming languages.		

Content: breakdown of the topics	**Contact hours**							**Self-study work: time and assignments**	
	Lectures	Tutorials	Seminars	Practice	Laboratory work (LW)	*Tutorial during LW*	**Contact hours**	**Self-study hours**	**Assignments**
1. Introduction to Computer Architecture	2				2	*8*	**4**	**4**	I. Assembler programming ABC. Presentation work I. II. Files treatment, various programming tasks Presentation work II. III. Interrupts programming, instructions recognition Presentation work III.
2. Computer structure	2				2		**4**	**4**	
3. CISC and RISC architecture. Microprogramming	6				2		**8**	**8**	
4. Memory organization architecture	2				2		**4**	**4**	
5. Memory addressing mechanism	2				2		**4**	**4**	
6. Instructions' formats. Assembler mnemonics	2				6		**8**	**8**	
7. Data formats	2				2		**4**	**4**	
8. Interruptions mechanism	2				2		**4**	**4**	
9. Computer instructions system	8				10		**18**	**12**	
10. Extended data formats and instructions	4				2		**6**	**6**	
11. Self-preparation and exam		2					**4**	**16**	2 hours for tutorial, 16 hours for preparation, 2 hours for exam
Total	**32**	**2**			**32**	*8*	**68**	**74**	

Assessment strategy	**Weight %**	**Deadline**	**Assessment criteria**
Exam. Evaluation of theoretical knowledge using open question exercises	60	Exam session	Performance of exercise is assessed in points 0 or 1. The points of 6 exercises are summed up. The assessment result is a value from 0 to 6.
Performance of laboratory works	40	7th, 12th and 16th week of semester	Three laboratory works are assessed. The weight of first work – 1 point. The weight of second and third works – 1,5 points. Minimal value of assessment that ensures the right to pass exam is 1 point, i.e. 25 % of assessment fully performed all laboratory works.

Author	**Publishing year**	**Title**	**Number or volume**	**Publisher or URL**
Required reading				
Antanas Mitašiūnas	2003	Computer architecture. Teaching book (in Lithuanian)		Vilnius, 126 p. http://www.mif.vu.lt/katedros/cs/Asmen/Kompiuteriu%20architektura.pdf
Peter Abel	1995	IBM PC Assembly Language and Programming		Prentice Hall International, Inc, Third Edition
Andrew S.Tanenbaum	1999	Structured computer organization		Prentice Hall PTR, Fourth Edition
Recommended reading				
Vytautas Urbanavičius	2004	Computers and their architecture. Teaching book. 1st part. (in Lithuanian)		Vilnius, "Technika"
Vytautas Urbanavičius	2004	Computers and their architecture. Teaching book. 2nd part. (in Lithuanian)		Vilnius, "Technika"