



RTU Course "Computer Organization and Assembly Language"

12306 Department of Applied Computer Science

General data

Code	DPI343
Course title	Computer Organization and Assembly Language
Course status in the programme	Compulsory/Courses of Limited Choice
Course level	Undergraduate Studies
Course type	Academic
Field of study	Computer Science
Responsible instructor	Uldis Sukovskis
Academic staff	Pāvels Rusakovs Gundars Alksnis
Volume of the course: parts and credits points	1 part, 3.0 Credit Points, 4.5 ECTS credits
Language of instruction	LV, EN
Annotation	This course covers organization principles of CPU, main memory, external devices and assembly language programming. Students will obtain skills and knowledge to operate computer components using low level programming techniques.
Goals and objectives of the course in terms of competences and skills	To obtain knowledge about organization principles of computers, understanding of operation and low level programming of computer components. To obtain skills in Assembly language programming and to apply these skills for development of specific software.
Structure and tasks of independent studies	To use literature sources, to prepare and execute individual tasks in computer lab.
Recommended literature	Kip R. Irvine. Assembly Language for Intel-Based Computers (5th Edition). – Prentice Hall, 2006. ISBN-13: 978-0132383103, 752 lpp. Peter Abel. IBM PC Assembly Language and Programming (5th Edition). – Prentice Hall, 2001. ISBN-13: 978-0130306555, 540 lpp. Peter Norton. Peter Norton's Computing Fundamentals (6 edition). – Career Education, 2004. ISBN-13: 978-0072978476, 608 lpp. Randall Hyde. The Art of Assembly Language (2nd Edition). - No Starch Press, 2010. ISBN-13: 978-1593272074, 760 lpp.
Course prerequisites	Numbering systems, Boolean algebra, C/C++ programming.

Course outline

Theme	Hours
Computer processors, registers, addressing of the main memory.	2
Compilation and execution of programs. Basics of Assembly language, programming of algorithms.	6
Operating system functions. Keyboard input and video output.	2
Programming of procedures, passing of parameters.	2
Interrupt handling. Hardware and software interrupts.	4
Programming of interrupt service routines.	4
Operations with computer components (keyboard, timer, video card, etc.).	12
Laboratory work.	16

Learning outcomes and assessment

Learning outcomes	Assessment methods
Understands the place and role of assembly programming in the context of other programming languages.	Passed tests and examination.
Is able to independently create and explain simple assembler language programs.	Successfully completed and defended laboratory works.
Is able to analyze the assembler programs for the structure and operation.	Passed tests and examination.
Understands and is able to explain the interrupt processing and operation of computer components (keyboard, video adapter, timer, etc.).	Passed tests and examination.

Study subject structure

Part	CP	Hours per Week			Tests		
		Lectures	Practical	Lab.	Test	Exam	Work
1.	3.0	2.0	0.0	1.0		*	