Course name : **Low-level programming**		
Field of study: **Computer science**	Type of study: **Full-time**	Course code : **D2_03**
Course characteristics: **Mandatory within the additional content**	Level: **I**	Year: **II** Semester: **IV**
Type of classes: **lectures, laboratories**	Hours per week: **1lec 2lab**	ECTS points: **3 ECTS**

COURSE GUIDE

AIMS

C1. Acquisition by the students with the history of processor, with their basic features, architecture and mechanisms used in them.

C2. Knowledge of chosen processor instructions and assembler directives.

C3. Acquisition by the students with the mechanisms and low-level programming methodology using selected examples.

C4. Acquisition by the students practical skills in the use of low-level programming systems, the use of processor instruction and writing programs in low-level language.

PREREQUISITES

1. Elementary knowledge of digital technology, computer architecture and programming basics.
2. Knowledge of work safety in the use of computer systems.
3. Ability to use different sources of information and technical documentation.
4. Ability to work independently and in a group.
5. Ability to correctly interpret and present their own activities.

LEARNING OUTCOMES

EK 1 – has basic theoretical knowledge of low-level programming,

EK 2 – knows the history and properties of processors,

EK 3 – have knowledge about architecture of processors,

EK 4 – knows the fixed point processor instructions, assembler directives and operators,

EK 5 – has the knowledge to write programs in assembler

EK 6 – knows the processor capabilities in the real calculations,

EK 7 – have knowledge about the SIMD instructions

EK 8 – able to use the packages to write programs or inserts in assembly language,

EK 9 – can use the processor instructions,

EK 10 – can implement in assembler structures known from high-level languages,

EK 11– has the ability to perform mathematical calculations in assembler,

EK 12 – able to prepare programs using SIMD instructions.

CONTENT

Lectures	Hours
W 1 – The history and properties of processors.	**1**
W 2 – Processor architecture.	**1**
W 3 – Addressing modes. Transfer instructions.	**1**
W 4 – Arithmetic instructions.	**1**
W 5 – Construction of program. Directives and Operators.	**1**
W 6 – Conditional and jump instructions.	**1**
W 7 – Logic, shift and rotation instructions.	**1**
W 8 – Operations on flags, bits and bytes.	**1**
W 9 – String and segments operations.	**1**
W 10 – Real types. Basic floating point operations.	**1**
W 11 – Transcendental function operations. Loading constants.	**1**
W 12 – Comparison and control operations.	**1**
W 13 – SIMD instructions - MMX.	**1**
W 14 – SIMD instructions - SSE.	**1**
W 15 – SIMD instructions - AVX.	**1**
LABORATORIES	**Hours**
L 1 – Software for writing programs in assembler.	**2**
L 2 – Simple subroutines. Running step by step.	**2**
L 3 – Loops and conditionals constructions.	**2**
L 4 – Operations on vectors.	**2**
L 5 – Working with matrices.	**2**
L 6 – Subroutines and using the stack.	**2**
L 7 – Operations on BCD numbers.	**2**
L 8 – String operations.	**2**
L 9 – Basic operations of the real numbers.	**2**
L 10 – Transcendental functions.	**2**
L 11 – Calculations using the real matrix.	**2**
L 12 – Application of comparison instructions of real numbers.	**2**
L 13 – Program using the SIMD instructions - MMX.	**2**
L 14 – Program using the SIMD instructions - SSE.	**2**
L 15 – Program using the SIMD instructions - AVX.	**2**

TEACHING TOOLS

1. – Lectures using multimedia presentations
2. – Examples of programs in assembler
3. – Laboratory guides
4. – Reports from laboratory activities
5. – Website with materials for the course

LITERATURE

1. Adam Błaszczyk: Win32ASM. Asembler w Windows, Helion 2004,
2. Randall Hyde: Asembler. Sztuka programowania, Helion 2004,
3. Stanisław Kruk: Asembler w koprocesorze, Mikom 2003,

4. Ryszard Goczyński, Michał Tuszyński: Mikroprocesory 80286, 80386 i i486, Komputerowa Oficyna Wydawnicza „HELP" 1991,
5. Michał Tuszyński, Ryszard Goczyński: Koprocesory arytmetyczne 80287 i 80387 oraz jednostka arytmetyki zmiennoprzecinkowej mikroprocesora i486, Komputerowa Oficyna Wydawnicza „HELP" 1992,
6. Intel® 64 and IA-32 Architectures Software Developer's Manual,
7. G.Syck, Turbo Assembler - Biblia użytkownika, LTP Oficyna Wydawnicza, 2002,
8. A.Rydzewski, Mikrokomputery jednoukładowe rodziny MCS-51,

TEACHERS

1. dr hab. inż. Jarosław Bilski, assoc. prof, jaroslaw.bilski@iisi.pcz.pl

2. dr inż. Piotr Dziwiński, piortr.dziwinski@iisi.pcz.pl

ADDITIONAL NOTES

1. Links to course unit teaching materials can be found on the http://iisi.pcz.pl/CIaDM/ website for current students. **http://www.iisi.pcz.pl/index.php/pl/do-pobrania?func=select&id=16/** .