#### card of course

|  |  |
| --- | --- |
| Subject name | Computer systems architecture |

1. The placement of the subject in the study system

|  |  |
| --- | --- |
| 1.1. Field of study | Computer science |
| 1.2. Form and path of study | Full-time/Part-time |
| 1.3. Level of education | First-cycle studies |
| 1.4. Study profile | Practical |

|  |  |
| --- | --- |
| 1. 5. Specialty | - |
| 1.6. Subject Coordinator | Dr Rafał Stęgierski |

2. General characteristics of the subject

|  |  |
| --- | --- |
| 2.1. Belonging to a subject group | Directional/Practical |
| 2.2. Number of ECTS | 4 |
| 2.3. Language of lectures | English |
| 2.4. Semesters in which the subject is taught | II |
| 2.5.Criteria for selecting course participants | - |

1. Learning outcomes and course delivery
	1. Subject Objectives

|  |  |
| --- | --- |
| No. | Subject Objectives |
|
| C1 | Learn key information about assemblers, mainly x86. |
| C2 | Learning the architecture of microcontrollers and how to program them. Familiarizing students with the process of data processing within microcontrollers. |
| C3 | Acquiring skills in using number systems and information coding methods. |
| C4 | Acquire skills in recognizing and explaining the basic elements of computer system architecture, such as the processor, memory, buses, input/output devices, and their interdependencies. |

* 1. Subject-specific learning outcomes, divided into knowledge , skills and competences , with reference to the directional learning outcomes

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Description of subject learning outcomes | Reference to directional effectslearning (symbols) | Method of implementation (mark "X") |
| ST | NST |
| Classes at the University | Activities on the platform | Classes at the University | Activities on the platform |
| After passing the course, the student knows and understands **the knowledge** |
| W1 | Has structured, theoretically based general knowledge of computer system architecture and hardware support for operating systems | INF\_W05INF\_W13INF\_W18INF\_W21 | X |  |  | X |
| W2 | Has structured and theoretically based detailed knowledge of low-level programming languages and hardware interfaces for human-computer communication | X |  |  | X |
| W3 | Has knowledge of development trends and the most important new achievements in the field of computer system architecture, computer system efficiency, low-level programming principles, problems of biological, optical and quantum computers | X |  |  | X |
| W4 | Knows the basic methods, techniques and tools used to solve simple IT tasks in the field of computer system construction, embedded systems, human-computer communication interfaces, software engineering | X |  |  | X |
| After passing the course, the student is **able** to: |
| U1 | The student is able to operate a processor simulator, test and analyze the operation of assembly code, identify errors and optimize the code during program execution. | INF\_U02 INF\_U03 INF\_U05 INF\_U07 | X |  | X |  |
| U2 | Is able to use information and communication techniques used in the implementation of IT projects during teamwork as part of workshop and laboratory activities | X |  | X |  |
| U3 | Can create assembler programs, including declaring data, stacks, arrays, procedures, implementing jump instructions, loops and bit operations | X |  | X |  |
| U4 | Can integrate assembly code with applications written in high-level languages, using WinAPI functions and creating assembly inline code. | X |  | X |  |
| After completing the course, the student is ready to take part in **social competences.** |
| K1 | Constantly improving their competences - they understand that in IT knowledge and skills become outdated very quickly | INF\_K02 INF\_K06 | X |  | X |  |
| K2 | Knows examples and understands the causes of malfunctioning systems that have led to serious financial or social losses or serious loss of health or even life in the field of construction and programming of microprocessor control systems | X |  | X |  |

3.3. Forms of teaching and their number of hours - Full-time studies (ST), Part-time studies (NST)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Path | Lecture | Exercises | Design | Workshop | Laboratory | Seminar | Lecturer | Classes conducted using distance learning methods and techniques in the form of a lecture | Other | **ECTS points** |
| **ST** | 15 |  |  |  | 30 |  |  |  |  | 4 |
| **NST** |  |  |  |  | 15 |  |  | 10 |  | 4 |

3.4. Content of education (separately for each form of classes: (W, ĆW, PROJ, WAR, LAB, LEK, OTHER). It should be marked (X) how the given content will be implemented (classes at the university or classes on the e-learning platform conducted using distance learning methods and techniques)

TYPE OF CLASS: LECTURE

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Content of the course | Reference to subject-specific learning outcomes | Method of implementation (mark "X") |
| ST | NST |
| **Classes at the University** | **Activities on the platform** | **Classes at the University** | **Activities on the platform** |
| 1. | Computers – structure and principle of operation. | W1-W4 | X |  |  | X |
| 2. | Construction and operation of a microprocessor | W1-W4 | X |  |  | X |
| 3. | Interrupts, memory, buses and devices. | W1-W4 | X |  |  | X |
| 4. | List of processor instructions. | W1-W4 | X |  |  | X |
| 5. | Arithmetic of digital machines. | W1-W4 | X |  |  | X |
| 6. | Data path. | W1-W4 | X |  |  | X |
| 7. | Controller module. | W1-W4 | X |  |  | X |
| 8. | Pipeline processing. | W1-W4 | X |  |  | X |
| 9. | Organizing Computer Memory | W1-W4 | X |  |  | X |
| 10. | Computer system architectures | W1-W4 | X |  |  | X |
| 11. | Summary of classes and discussion of grades |  | X |  |  | X |

TYPE OF CLASS: LABORATORY

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Content of the course | Reference to subject -specific learning outcomes | Method of implementation (mark "X") |
| ST | NST |
| **Classes at the University** | **Activities on the platform** | **Classes at the University** | **Activities on the platform** |
| 1. | Coding numbers and texts | U3 | X |  | X |  |
| 2. | Getting to know the operation of the processor simulator. | U1 | X |  | X |  |
| 3. | Basics of creating applications in assembler - sections, data declarations, stack, arrays, procedures, jump statements, loops, bit operations | U3 | X |  | X |  |
| 4. | Using WinAPI functions - displaying data, simple control instructions, using macro definitions | U4 | X |  | X |  |
| 5. | Creating assembly inline code in high-level languages | U4 | X |  | X |  |
| 6. | Building a computer, installing operating systems | K1, K2 | X |  | X |  |
| 7. | Summary of classes and discussion of grades. |  | X |  | X |  |

3.5. Methods of verifying learning outcomes (indication and description of methods of conducting classes and verification of achievement of learning outcomes and method of documentation)

Lecture credit – test exam

Passing the exercises:

- 60% of the grade - laboratory exercises performed based on the instructions provided, introducing the principles of coding numbers and texts and understanding the principles of programming in low-level languages:

1. Coding numbers and texts
2. Getting to know the operation of the processor simulator.
3. Basics of creating applications in assembler - sections, data declarations, stack, arrays, procedures, jump statements, loops, bit operations
4. Using WinAPI functions - displaying data, simple control instructions, using macro definitions
5. Creating assembly inline code in high-level languages

- 40% of the grade - presentation made in two-person groups on the construction and principles of computer operation and installation of operating systems (topic agreed with the instructor). The following are assessed: consistency of the study with the guidelines included in the instructions, substantive correctness and relevance of the issues presented.

|  |  |  |  |
| --- | --- | --- | --- |
| Subject Effects | Teaching methods | Methods of verifying learning outcomes | Documentation methods |
| KNOWLEDGE |
| W1-W4 | Complex multimedia presentation, problem solving | Test exam – passing the lecture | Test archived on the platform |
| SKILLS |
| U1-U4 | Laboratory, partial projects, discussion on the solutions used | Completion of 5 sets of laboratory tasks – 60% of the laboratory gradePreparation of selected issues in the form of a presentation – 40% of the laboratory grade | The content of the tasks and reports of the completed laboratory tasks are posted on the platform ,Prepared presentations posted on the platform |
| SOCIAL COMPETENCES |
| K1-K2 | Laboratory, partial projects, discussion on the solutions used | Completion of 5 sets of laboratory tasks – 60% of the laboratory gradePreparation of selected issues in the form of a presentation – 40% of the laboratory grade | The content of the tasks and reports of the completed laboratory tasks are posted on the platform,Prepared presentations posted on the platform |

3.6. Assessment criteria for the achieved learning outcomes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Learning effect | For a grade of 3 or "pass."the student knows and understands/is able to/is ready to | For a grade of 3.5, the student knows and understands/is able to/is ready to | For a grade of 4, the student knows and understands/is able to/is ready to | For a grade of 4.5, the student knows and understands/is able to/is ready to | For a grade of 5, the student knows and understands/is able to/is ready to |
| W | 51-60% of knowledge indicated in learning outcomes | 61-70% of knowledge indicated in learning outcomes | 71-80% of knowledge indicated in learning outcomes | 81-90% of knowledge indicated in learning outcomes | 91-100% of knowledge indicated in learning outcomes |
| U | 51-60% of skills indicated in learning outcomes | 61-70% of skills indicated in learning outcomes | 71-80% of skills indicated in learning outcomes | 81-90% of skills indicated in learning outcomes | 91-100% of skills indicated in learning outcomes |
| K | 51-60% of skills indicated in learning outcomes | 61-70% of skills indicated in learning outcomes | 71-80% of skills indicated in learning outcomes | 81-90% of skills indicated in learning outcomes | 91-100% of skills indicated in learning outcomes |

3.7. Literature

**Basic**

1. Lacamera Daniele; Embedded Systems Architecture: Explore architectural concepts, pragmatic design patterns, and best practices to produce robust systems; Packt Publishing; Birmingham 2018
2. W. Stallings, z ang. przeł. Jacek B. Szporko, Organizacja i architektura systemu komputerowego: projektowanie systemu a jego wydajność. Wydanie trzecie zmienione i rozszerzone, Wydawnictwa Naukowo-Techniczne, Warszawa 2004
3. Metzger P., Anatomia PC. Architektura komputerów zgodnych z IBM PC. Kompendium. Wiedza o architekturze komputerów PC w pigułce, Helion, 2008

**Supplementary**

1. Wróbel Eugeniusz, Asembler 8086/88
2. Kruk Stanisław, Turbo Asembler : idee, polecenia, rozkazy procesora Pentium, Mikom, Warszawa, 2000
3. Kruk Stanisław, Język Assembler dla początkujących

4. Student workload - ECTS points balance

|  |  |
| --- | --- |
| **Types of student activity** | **Student Load** |
| **ST** | **NST** |
| **Classes requiring direct contact between the student and the academic teacher at the university premises** | **45** | **25** |
| Classes included in the study plan | 45 | 25 |
| **Student's own work** | **55** | **75** |
| Ongoing preparation for classes, preparation of project work/presentations/etc. | 25 | 35 |
| Preparation for passing classes | 30 | 40 |
| **TOTAL STUDENT HOURLY LOAD** | **100** | **100** |
| **Number of ECTS points** | **4** | **4** |

|  |  |
| --- | --- |
| Last change date | 30/09/2024 |
| The changes were introduced | INF Education Quality Team |
| The changes were approved | Arkadiusz Gwarda, M.A. |