POZNAN UNIVERSITY OF TECHNOLOGY

EUROPEAN CREDIT TRANSFER AND ACCUMULATION SYSTEM (ECTS)

pl. M. Skłodowskiej-Curie 5, 60-965 Poznań

COURSE DESCRIPTION CARD - SYLLABUS

Course name

Programming languages

Course

Field of study

Mechatronics

Area of study (specialization)

Level of study

First-cycle studies

Form of study

full-time

Year/Semester

2/4

Profile of study

general academic

Course offered in

polish

Requirements

compulsory

Number of hours

Lecture	Laboratory classes	Other (e.g. online)
0	0	0
Tutorials	Projects/seminars	
15	15	

Number of credit points

2

Lecturers

Responsible for the course/lecturer:

dr hab. inż. Wojciech Pietrowski

email: wojciech.pietrowski@put.poznan.pl

Wydział Automatyki, Robotyki i Elektrotechniki

ul. Piotrowo 3A, 60-965 Poznań, room 651

Responsible for the course/lecturer:

Prerequisites

Basic knowledge of computer science, computer hardware, operating systems, algorithms and data structures. Operating computer, MS Windows operating system and application software.

Course objective

Provide basic knowledge of programming languages, mastering the ability to develop algorithms and the basics of structured programming in C / C ++.

Course-related learning outcomes

Knowledge

The student knows the basic concepts of programming languages, algorithms, data structures and the development of computer software.

The student is able to apply basic programming techniques and use the selected programming environment.

The student is able to apply the methods of creating algorithms and computer software to solve selected tasks and technical problems.

Skills

The student knows how to formulate and solve tasks and technical problems using engineering analytical and simulation methods.

The student knows how to algorithmize tasks and selected engineering problems and develop corresponding computer software in C / C ++.

The student knows how to test the developed software and assess their correct functioning.

Social competences

The student understands the need for lifelong learning; can inspire and organize the learning process of other people.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lecture: credit on the basis of a test consisting of general and test questions. Rating scale 51-60% points - satisfactory, 61-70% points satisfactory+, 71-80% points - good, 81-90% points - good +, 91-100% points - very good.

Laboratory: rewarding practical knowledge gained during previous laboratory exercises, checking practical programming skills in C ++ (final test), assessment of knowledge and skills related to the implementation of individual and group programming projects.

Obtaining additional points for activity during classes, especially for: the ability to cooperate as part of a team practically carrying out a detailed task in the laboratory, the use of elements and techniques that go beyond the material of the lecture and laboratory exercises, aesthetic diligence of completed projects.

Programme content

Lecture:

Division of programming languages. Compilation, interpretation. Low and high level languages. Programming languages overview: Assembler, Basic, Fortran, C, C ++, C #, Pascal, JAVA, PHP. Development environments. Source program editors and compilers, Debugger. Structural programming. Basics of programming in Pascal and Object Pascal, C / C ++.

Laboratory:

Software development using conditional instructions, iterative instructions. Using libraries with functions. Development of own functions and libraries.

Teaching methods

Lecture: multimedia presentation and presentation of writing and executing selected software directly in C / C ++.

Laboratory: practical exercises on the elements of the C / C ++ language, writing GUI software in this language.

Bibliography

Basic

Stroustrup Bjarne, Programowanie : teoria i praktyka z wykorzystaniem C++, Helion 2020

Grębosz Jerzy, Opus Magnum C++11 : programowanie w języku C++, Helion 2020

Zieliński Józef, Podstawy programowania w języku C++, Oficyna Wydawnicza "Impuls", 2013

Additional

Grębosz Jerzy, Pasja C++ , Oficyna Kallimach 2004

Breakdown of average student's workload

	Hours	ECTS
Total workload	50	2,0
Classes requiring direct contact with the teacher	32	1,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation)[1]	18	1,0

[1] delete or add other activities as appropriate