



COURSE DESCRIPTION CARD - SYLLABUS

Course name

Object oriented programming

Course

Field of study

Mechatronics

Area of study (specialization)

Level of study

First-cycle studies

Form of study

full-time

Year/Semester

3/5

Profile of study

general academic

Course offered in

polish

Requirements

compulsory

Number of hours

Lecture

15

Laboratory classes

15

Other (e.g. online)

0

Tutorials

0

Projects/seminars

0

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, operating systems, algorithms and data structures, programming languages. Windows operating system and the ability to structured programming in C ++, Pascal.

Course objective

Getting to know the theoretical and practical issues related to object-oriented programming in C ++.

Course-related learning outcomes

Knowledge

The student knows the basic concepts of object-oriented programming. The student knows the basic object-oriented programming languages: C ++, Java. The student knows the basic techniques of object-oriented programming in selected programming environments. The student knows the program structure, module structure, selected visual components.

Skills

The student is able to obtain information from technical literature and the Internet regarding object-



oriented programming. The student is able to define and use the class type, function templates, inheritance technique. The student is able to build computer software in integrated environments using the object-oriented programming technique. The student is able to test the developed software and assess the correctness of their 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

C++, Java. Object-oriented programming in the Microsoft Visual Studio environment. Source program editors. Debugger. Object oriented programming. Classes, objects, inheritance, data encapsulation, polymorphism and virtual methods, abstract classes, function templates, class templates. Creation and processing of objects, objects as arguments of functions. Components, forms, properties, events, exception handling. Using object-oriented programming to simulate selected operating states of automation actuators.

Teaching methods

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

Laboratory exercises: 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