



Subject card

Subject name and code	Object-oriented programming, PG_00054485						
Field of study	Automation, Robotics and Control Systems						
Date of commencement of studies	February 2023	Academic year of realisation of subject			2023/2024		
Education level	second-cycle studies	Subject group					
Mode of study	Full-time studies	Mode of delivery			at the university		
Year of study	2	Language of instruction			Polish		
Semester of study	3	ECTS credits			4.0		
Learning profile	general academic profile	Assessment form			assessment		
Conducting unit	Faculty of Electrical and Control Engineering						
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Paweł Kowalski					
	Teachers	dr inż. Paweł Kowalski					
Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	15.0	15.0	0.0	60
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM		
	Number of study hours	60	12.0	28.0	100		
Subject objectives	Understanding the basic concepts of object-oriented programming. Acquiring the skills of object modelling, solving problems using objects and relations between them. Acquiring the ability to build and run Java, Python and Kotlin programs with object-oriented programming methods. Acquisition of the ability to design and build a graphical user interface.						
Learning outcomes	Course outcome	Subject outcome			Method of verification		
	K7_U03	The student is able to prepare and present a presentation of an implemented engineering task.			[SU5] Assessment of ability to present the results of task		
	K7_W11	The student has in-depth knowledge of object-oriented application design.			[SW3] Assessment of knowledge contained in written work and projects [SW2] Assessment of knowledge contained in presentation [SW1] Assessment of factual knowledge		
	K7_U04	The student works independently looking for solutions to the problems encountered in the documentation and on internet forums. Identifies and removes the causes of application malfunctions. Gathering the information necessary to complete the project.			[SU5] Assessment of ability to present the results of task [SU3] Assessment of ability to use knowledge gained from the subject [SU1] Assessment of task fulfilment		
	K7_W06	The student has an extensive knowledge of object-oriented application design.			[SW3] Assessment of knowledge contained in written work and projects [SW2] Assessment of knowledge contained in presentation [SW1] Assessment of factual knowledge		
	K7_U07	The student is able to use analytical, simulation and experimental methods to formulate and solve engineering tasks.			[SU5] Assessment of ability to present the results of task [SU4] Assessment of ability to use methods and tools [SU1] Assessment of task fulfilment		

Subject contents	Introduction to object oriented programming and Java. Java data types. Support for standard input and output. Classes and objects. Defining a class and an object. Definitions of fields, methods and constructors. The life cycle of objects and the garbage collector mechanism. Access modifiers visibility of class members. Data Hermetization. Operations on arrays. Inheritance and polymorphism. File handling. Catching, handling and throwing exceptions. Building a GUI application. The essence of event programming. Designing and building mobile applications using the Kotlin language. Data acquisition using an Internet robot.		
Prerequisites and co-requisites	Basic programming skills		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	laboratory tasks	50.0%	15.0%
	exam	50.0%	40.0%
	project	50.0%	45.0%
Recommended reading	Basic literature	<ul style="list-style-type: none"> • B. Eckel, Thinking in Java, Helion. • Java Programming Language, Decodejava , https://www.decodejava.com. • S. Ludwiczak, M. Kunert: Kurs Programowania Java od Podstaw. https://javastart.pl/baza-wiedzy,JavaStart, 2021. • Java Technical Details, http://java.sun.com. 	
	Supplementary literature	<ul style="list-style-type: none"> • C. S. Horstmann, G. Cornell: Java. Techniki zaawansowane. Helion, Gliwice 2009. • A. Redko: Advanced Java Preparing you for Java Mastery, 2015. 	
	eResources addresses	Adresy na platformie eNauczenie:	
Example issues/ example questions/ tasks being completed	<ul style="list-style-type: none"> • Development of a simple expert system. • Implementation of the game of life according to the principles of John Conway. • Designing classes for the passage of a computer game. • Development of a graphic interface for the selected application. • Handling of events generated in the designed graphical interface. • Tic-tac-toe game for mobile devices. • Development of an Internet robot 		
Work placement	Not applicable		