

Subject card

Subject name and code	Programming languages I, PG_00155294						
Field of study	Informatics						
Date of commencement of studies	October 2025		Academic year of realisation of subject			2025/2026	
Education level	Bachelor's studies		Subject group			Obligatory subject group in the field of study Subject group related to practical vocational preparation	
Mode of study	full-time studies		Mode of delivery			at the university	
Year of study	1		Language of instruction			Polish	
Semester of study	2		ECTS credits			4.0	
Learning profile	practical		Assessment form				
Conducting unit	Institute of Informatics -> Faculty of Mathematics, Physics and Informatics -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		dr hab. Wiesław Pawłowski				
	Teachers						
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	30.0	0.0	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		0.0		40.0	100
Subject objectives	The aim of the course is to familiarize students with the most important mechanisms offered by modern programming languages, including elements of functional, object-oriented and asynchronous programming.						
Learning outcomes	Course outcome		Subject outcome			Method of verification	
	[INFL3_U04] can create, run and test programs using dedicated tools and design patterns		is able to use basic tools - programming editor, compiler and simple build tool.			[SU5] implementation of a problem task	
	[INFL3_U09] is able to - in accordance with the given specification - design and implement IT system		knows how to match learned programming paradigms and mechanisms to problem specifications			[SU4] test/exam - oral or written [SU5] implementation of a problem task	
	[INFL3_W04] has ordered, theoretically founded knowledge in programming, algorithms and complexity, programming languages and paradigms		knows the basic features of the functional, object-oriented and asynchronous programming models			[SW4] test/exam - oral or written [SW5] implementation of a problem task	
	[INFL3_U08] assesses the usefulness of various paradigms and related programming tools to solve various types of problems		knows how to solve programming problems with the help of learned programming paradigms			[SU4] test/exam - oral or written [SU5] implementation of a problem task	
	[INFL3_K02] can precisely formulate questions to deepen his/ her own understanding of a given topic or to find missing elements of reasoning		can analyze simple programming problems			[SK4] test/exam - oral or written [SK5] implementation of a problem task	

Subject contents	The course covers programming paradigms and the most important concepts offered by modern programming languages. • Functional programming: non-mutability, functions (including functions as values), tail recursion, polymorphism (type parameterization). • Collections as extended, parameterized data structures - the most important types and operations provided. • Basic concepts of object-oriented programming - objects, classes and interfaces/traits, inheritance mechanisms. • Elements of asynchronous programming In addition to the above-mentioned language mechanisms, the course will also discuss/present basic tools to help create, test and run programs.		
Prerequisites and co-requisites	none		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	programming colloquium	51.0%	100.0%
Recommended reading	Basic literature	• Lecture notes (slides and code examples). • M. Odersky, L. Spoon, B. Venners, F. Sommers, Programming in Scala, Fifth Edition, Artima Press, 2021. • F. Lopez-Sancho, Akka in Action, Second Edition, Manning 2023.	
	Supplementary literature	• D. Wampler, Programming Scala, Third Edition. O'Reilly Media, 2021.	
	eResources addresses	Supplementary https://pekko.apache.org/docs/pekko/current/actors.html - Apache Pekko – Classic Actors https://docs.scala-lang.org/getting-started/install-scala.html - How to install Scala and start writing, compiling and executing code. https://docs.scala-lang.org/scala3/book/introduction.html - Scala 3 introduction	
Example issues/ example questions/ tasks being completed			
Work placement	Not applicable		

Document generated electronically. Does not require a seal or signature.