University of Gdańsk

Subject card

Subject name and code	Programming Languages I, PG_00204160		
Field of study	Informatics		
Date of commencement of studies	October 2026	Academic year of realisation of subject	2026/2027
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	credit
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
[INFPL3_U04] is able to use the acquired knowledge when creating, running and testing 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
[INFPL3_U08] is able to assess the usefulness of various paradigms and programming tools for solving 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
[INFPL3_K02] is ready to recognize the importance of knowledge in solving cognitive problems and practical and seeking opinions experts in case of difficulties with independent problem solving	can analyze simple programming problems, is ready to use technical documentation, professional literature and expert sources	[SK4] test/exam - oral or written [SK5] implementation of a problem task
[INFPL3_W04] knows and understands advanced issues in programming, algorithms and computational complexity, programming languages and paradigms, as well as the complex relationships between these areas	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
[INFPL3_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

<table>
<tr><td>Subject contents</td><td>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.</td></tr>
<tr><td>Prerequisites and co-requisites</td><td>none</td></tr>
<tr><td>Assessment methods and criteria</td><td><table><tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr><tr><td>programming colloquium</td><td>51.0%</td><td>100.0%</td></tr></table></td></tr>
<tr><td>Recommended reading</td><td><table><tr><td>Basic literature</td><td>• 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.</td></tr><tr><td>Supplementary literature</td><td>• D. Wampler, Programming Scala, Third Edition. O'Reilly Media, 2021.</td></tr><tr><td>eResources addresses</td><td></td></tr></table></td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td></td></tr>
<tr><td>Work placement</td><td>Not applicable</td></tr>
</table>

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