

Subject card

Subject name and code	Programming in Rust, PG_00179316						
Field of study	Programowanie w języku Rust						
Date of commencement of studies	October 2023		Academic year of realisation of subject			2025/2026	
Education level	Bachelor's studies		Subject group			Obligatory subject group in the field of study Optional subject group Subject group related to scientific research in the field of study	
Mode of study	full-time studies		Mode of delivery			at the university	
Year of study	3		Language of instruction			Polish Polish/English	
Semester of study	5		ECTS credits			3.0	
Learning profile	academic		Assessment form			credit	
Conducting unit							
Name and surname of lecturer (lecturers)	Subject supervisor		dr Piotr Arłukowicz				
	Teachers		dr Piotr Arłukowicz				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	15.0	0.0	0.0	30
	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	30		0.0		0.0	30
Subject objectives	The objective of the course is to familiarize students with the fundamentals of programming in Rust as well as the use of the Cargo ecosystem.						
Learning outcomes	Course outcome		Subject outcome			Method of verification	
	[INFL3_W05] has general knowledge of various programming paradigms and programming languages; has detailed knowledge of object-oriented design and programming methods and patterns		The student understands the operation of the borrow checker and other advanced features of Rust			[SW2] prezentacja/projekt/referat/ raport	
	[INFL3_U06] is able to design, create, run and test programs using dedicated tools and adequate templates		The student is capable of building a project using the Cargo ecosystem that addresses the given problem.			[SU2] prezentacja/projekt/referat/ raport	
	[INFL3_U03] can design and analyze algorithms for their correctness and computational complexity using appropriate algorithmic techniques and data structures		The student is capable of developing algorithm implementations that make use of the advanced features of Rust.			[SU2] prezentacja/projekt/referat/ raport	

Subject contents	Introduction to Rust Programming: First Program and Cargo Primitive Data Types Compound Data Types Error Handling The Borrow Checker Ownership Method Syntax and Implementations Smart Pointers and Heap Allocation Object-Oriented Programming Concepts in Rust		
Prerequisites and co-requisites	Desired is the knowledge about other programming language, C++ is beneficial.		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	Presence	0.0%	26.0%
	Project	51.0%	74.0%
Recommended reading	Basic literature	https://doc.rust-lang.org/book/	
	Supplementary literature	https://doc.rust-lang.org/rust-by-example/	
	eResources addresses		
Example issues/ example questions/ tasks being completed	1. Development of a TODO application. 2. Design and implementation of a forest fire simulation program. 3. Solving layout problems in container systems using the Flexbox model.		
Work placement	Not applicable		

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