

**Subject card**

<b>Subject name and code</b>	Programming Languages, PG_00143841						
<b>Field of study</b>	Informatics						
<b>Date of commencement of studies</b>	October 2024	<b>Academic year of realisation of subject</b>			2024/2025		
<b>Education level</b>	Bachelor's studies	<b>Subject group</b>		Obligatory subject group in the field of study			
<b>Mode of study</b>	part-time studies	<b>Mode of delivery</b>			at the university		
<b>Year of study</b>	1	<b>Language of instruction</b>			Polish polish		
<b>Semester of study</b>	1	<b>ECTS credits</b>			12.0		
<b>Learning profile</b>	academic	<b>Assessment form</b>					
<b>Conducting unit</b>							
<b>Name and surname of lecturer (lecturers)</b>	<b>Subject supervisor</b>		dr Maciej Dziemiańczuk				
	<b>Teachers</b>		dr Maciej Dziemiańczuk mgr Mateusz Miotk mgr Michał Zakrzewski				
<b>Lesson types</b>	<b>Lesson type</b>	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	<b>Number of study hours</b>	40.0	0.0	40.0	0.0	0.0	80
	E-learning hours included: 0.0						
<b>Learning activity and number of study hours</b>	<b>Learning activity</b>	Participation in didactic classes included in study plan		Participation in consultation hours		Self-study	SUM
	<b>Number of study hours</b>	80		0.0		220.0	300
<b>Subject objectives</b>	The goal is to teach students programming in the C language and to understand C code written by other programmers.						
<b>Learning outcomes</b>	<b>Course outcome</b>		<b>Subject outcome</b>		<b>Method of verification</b>		
	[INFL3_W05] has general knowledge of various programming paradigms and programming languages; has detailed knowledge of object-oriented design and programming methods and patterns		Distinguishes syntactic errors in program code from logical errors. Understands short code fragments. Knows the basic constructs of the C language. Knows what the standard libraries of the language are and what they contain.		[SW4] test/exam - oral or written [SW1] oral statement/ conversation/discussion		
	[INFL3_U09] is able to assess the suitability of programming paradigms and tools for solving problems of various types		In their programs, can use existing libraries and code written by others.		[SU3] text preparation/written work [SU5] implementation of a problem task [SU6] demonstration of practical skills		
	[INFL3_U06] is able to design, create, run and test programs using dedicated tools and adequate templates		Can write a program in the C language to solve a specific problem. Can implement a simple algorithm in the C language.		[SU3] text preparation/written work [SU5] implementation of a problem task [SU6] demonstration of practical skills		

Subject contents	1. Introduction to programming in the C language. Running simple programs. Syntax and semantics of a program. 2. Programming environment in the UNIX system. Compiling and linking a program using the gcc compiler. 3. Types, operators, and expressions. Declarations. Type conversion. Conditional expressions. Order of operations. 4. Control statements. Statements and blocks. 5. Functions and program structure. Scope of names. Header files. 6. Pointers and arrays. Program call arguments. Function pointers. 7. Structures. Typedef declarations. Unions. 8. Input and output. Overview of standard libraries: input/output, string handling functions.		
Prerequisites and co-requisites	no requirements		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	exam	50.0%	50.0%
	tests	50.0%	50.0%
Recommended reading	Basic literature	1. Brian W. Kernighan, Dennis M. Ritchie. Język ANSI C. WNT 2. Stephen Prata. Język C. Szkoła programowania, Helion 3. Keith Cooper, Linda Torczon. Engineering a Compiler, MORGAN KAUFMANN	
	Supplementary literature	not applicable	
	eResources addresses	Adresy na platformie eNauczanie:	
Example issues/ example questions/ tasks being completed	not applicable		
Work placement	Not applicable		

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