GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Introduction to Programming, PG_00067984						
Field of study	Biomedical Engineering						
Date of commencement of studies	October 2025		Academic year of realisation of subject	2025/2026			
Education level	first-cycle studies		Subject group	Obligatory subject group in the field of study			
Mode of study	Full-time studies		Mode of delivery	at the university			
Year of study	1		Language of instruction	Polish			
Semester of study	1		ECTS credits	2.0			
Learning profile	general academic profile		Assessment form	assessment			
Conducting unit	Department of Biomedical Engineering -> Faculty of Electronics Telecommunications and Informatics -> Wydziały Politechniki Gdańskiej						
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Magdalena Mazur-Milecka					
	Teachers	dr inż. Magdalena Mazur-Milecka					
Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	20.0	0.0	15.0	0.0	0.0	35
	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	35		5.0	10.0		50
Subject objectives	The aim of the course is to familiarize students with the fundamental concepts of structural and procedural programming using the C language, covering both theoretical foundations and practical implementation aspects.						

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K6_U04] can apply knowledge of programming methods and techniques as well as select and apply appropriate programming methods and tools in computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study	The student is able to apply basic programming methods and techniques in the C language to independently develop simple computer programs, including selecting appropriate language constructs, organizing the code structure, and using programming tools for compiling, testing, and debugging source code.	[SU1] Assessment of task fulfilment
[K6_W04] knows and understands, to an advanced extent, the principles, methods and techniques of programming and the principles of computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study, and organisation of systems using computers or such devices	The student knows and understands the principles and techniques of structural and procedural programming in the C language, including program structure, memory organization, data types, input–output operations, and code organization using functions. The student understands the basic principles of developing data-processing programs and the functioning of programming support tools (compiler, debugger).	[SW1] Assessment of factual knowledge
[K6_U12] can analyze the operation of components, circuits and systems related to the field of study, as well as measure their parameters and examine technical specifications, and plan and conduct experiments related to the field of study, including computer simulations and measurements, and interpret obtained results and draw conclusions	The student is able to plan and conduct tests of programs written in the C language, analyze their behavior based on program inputs and outputs, identify logical and runtime errors, and interpret the results. The student can carry out simple computational experiments (e.g., algorithm testing), draw conclusions based on observations, and modify the code to improve its performance.	[SU1] Assessment of task fulfilment

Subject contents

Lecture

1. Problem solving through programming
2. Algorithm vs. computer program. Programming languages. Structure of a program
3. Computer memory and data types. Variables and constants. Operators
4. Control flow: conditional statements, iterative statements
5. Basic interaction with the computer: input and output operations
6. Structure and purpose of functions. Basic usage of functions. Function libraries
7. String operations. File operations
8. Arrays
9. Memory management
10. Data structures
11. Pointers and their applications
12. Graphical user interfaces introduction

Laboratory

1. Introduction, Variables, Operators 2. Control Structures and Functions 3. Arrays and File Operations
4. Structures and Pointers

Prerequisites and co-requisites

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
final test	50.0%	40.0%
laboratory	50.0%	60.0%

Recommended reading

Basic literature

1. *The C Programming Language*, 2nd Edition commonly referred to as *ANSI C,* Brian W. Kernighan, Dennis M. Ritchie
2. *C Programming Absolute Beginners Guide*, 3rd Edition, Greg Perry, Dean Miller

	Supplementary literature	1. Understanding and Using C Pointers by Richard M Reese https://www.oreilly.com/library/view/understanding-and-using/9781449344535/ 2. Modern-C https://gustedt.gitlabpages.inria.fr/modern-c/
	eResources addresses	
Example issues/ example questions/ tasks being completed		
Work placement	Not applicable	

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