GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Basics of Programming, PG_00067422		
Field of study	Automatic Control, Cybernetics and Robotics		
Date of commencement of studies	October 2026	Academic year of realisation of subject	2026/2027
Education level	first-cycle studies	Subject group	Obligatory subject group in the field of study 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	1	Language of instruction	Polish
Semester of study	1	ECTS credits	5.0
Learning profile	general academic profile	Assessment form	assessment
Conducting unit	Department of Algorithms and Systems Modelling -> Faculty of Electronics Telecommunications and Informatics -> Faculties of Gdańsk University of Technology		
Name and surname of lecturer (lecturers)	Subject supervisor	dr Paweł Obszarski	
	Teachers	dr Paweł Obszarski dr inż. Aleksander Mroziński mgr inż. Andrzej Jastrzębski dr hab. inż. Iwona Kochańska dr inż. Joanna Raczek dr inż. Marcin Strąkowski dr inż. Andrzej Marczak dr hab. inż. Robert Janczewski dr inż. Katarzyna Karpienko dr inż. Maciej Wróbel dr inż. Bartłomiej Dec mgr inż. Jan Glinko	

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	10.0	55.0	125

Subject objectives	The course introduces computer programming. Students become familiar with basic programming techniques in C++. They learn to solve programming problems and the basics of algorithms.

Learning outcomes

Course outcome	Subject outcome	Method of verification
[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 develop, test and investigate simple computer programs, including those that simulate processes.	[SU1] Assessment of task fulfilment
[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 analyzes and solves basic programming problems. The student uses high-level programming language constructs that are syntactically and semantically correct. He writes and modifies programs in a high level language. Identifies and removes the causes of program malfunctions. He explains and can apply basic programming techniques.	[SU1] Assessment of task fulfilment [SU4] Assessment of ability to use methods and tools
[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 the constructs of a high-level programming language. He is able to interpret and analyze programs written in a high-level programming language. The student knows the rules for selection of programming methods and language constructs.	[SW1] Assessment of factual knowledge

Subject contents

Course content – lecture
1. introduction
2. variables - basic types and basic arithmetic, bitwise operations
3. logical operators, bool variables, conditional statement,
4. for, while, do while loops
5. arrays, array, vecotor,
6. string, c-string, cstring,
7. functions, passing by value, returning, learning effects.
8. modules,
9. pointers,
10. functions 2. recursion
11 structures
12. objects
13. stl
14. files
15. basic data structures

Prerequisites and co-requisites

Knowledge and skills in mathematics at secondary school level.

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
Programming lab assignments	51.0%	50.0%
Final test	51.0%	50.0%

Recommended reading

Basic literature	[1] Gręboszz J., Symfonia C++ Standard (tom 1 i 2), Editions 2000, Krakow 2008. [2] Podstawy programowania - materiały do wykładu.
Supplementary literature	Cormen Thomas H. Wprowadzenie do algorytmów
eResources addresses	

Example issues/ example questions/ tasks being completed

1. Write a program that reads an array of integers and rearranges its elements so that even numbers are at the beginning.
2. Write a function that calculates the number of divisors of an integer.

Practical activites within the subject

Not applicable