



COURSE DESCRIPTION

Course code	full-time studies	
	part-time-studies	
Course name	Podstawy Programowania 1	
Course name in English	Fundamentals of Programming 1	
Valid from academic year	2022/23	

PLACEMENT IN THE TEACHING PROGRAM

Field of study	Computer Science
Level of education	1st degree
Studies profile	General
Form and method of teaching classes	Full-time and part-time studies
Specialization	All specializations / Information systems / Computer graphics / Information and communication technology
Organizational unit responsible for the course	Department of Information Systems
Course coordinator	Arkadiusz Chrobot, PhD
Approved by	Dean of the Faculty of Electrical Engineering, Automatic Control and Computer Science Roman Deniziak, KUT prof., DSc, PhD

GENERAL CHARACTERISTIC OF THE COURSE

Course affiliation	Introductory Course	
Course status	Mandatory	
Language	English	
Semester	full-time studies	1st semester
	part-time-studies	1st semester
Requirements	No requirements	
Exam (YES/NO)	NO	
ECTS	5	

Course form		lecture	classes	laboratory	project	other
Hours per semester	full-time studies	30		30		
	part-time-studies	18		18		

LEARNING RESULTS

Category	Result Symbol	Learning Results	References to the field of study results
Knowledge	W01	The student knows the concepts of imperative programming, procedural programming and algorithm.	INF1_W07
	W02	The student knows the basic elements of a high-level programming language syntax (such as conditional and loop statements).	INF1_W07
	W03	The student knows basic data types, data structures and selected algorithms that use them.	INF1_W07
Skills	U01	The student is able to create basic program using the imperative programming paradigm.	INF1_U07
	U02	The student is able to apply basic elements of high-level programming language syntax in a program.	INF1_U07
	U03	The student is able to apply basic data types and data structures in a program.	INF1_U07
	U04	The student is able to use an Integrated Development Environment, to edit, compile and debug a program.	INF1_U07
Social competence	K01	The student understands the need of constant development of her/his programming skills.	INF1_K1
	K02	The student understands the risks resulting from lack of adequate skills of people who create software individually or in a team.	INF1_K2

COURSE CONTENT

Course Form	Content
lecture	<ol style="list-style-type: none"> 1. Introduction – concepts of imperative programming, procedural programming, algorithm; examples of algorithms; basic data types; variables and constants. 2. Initialization of variables, operators and expressions, basic input/output. 3. Control statements (conditional statement, switch statement, loops). 4. Functions, local variables, parameters. 5. Enumerations and linear arrays, processing of linear arrays. 6. Strings, processing of strings. 7. Multidimensional arrays, processing of multidimensional arrays. 8. Structures, unions and bit fields. 9. Files, processing of files. 10. Libraries and macros.
laboratory	<ol style="list-style-type: none"> 1. Introduction - the Integrated Development Environment. 2. Basic concepts: variables, constants, data types, basic input/output, operators, expressions. 3. Control statements. 4. Functions. 5. Enumerations and linear arrays. 6. Strings. 7. Multidimensional arrays. 8. Structures and unions. 9. Files. 10. Libraries and macros.

LEARNING RESULTS VERIFICATION METHODS

Result Symbol	Learning results verification methods					
	Oral Exam	Written Exam	Midterm	Project	Report	Other
W01			X			X
W02			X			X
W03			X			X
U01			X			X
U02			X			X
U03			X			X
U04			X			X
K01			X			X
K02			X			X

ASSESSMENT FORMS AND CRITERIA

Course Form	Assessment Form	Assessment Criteria
lecture	Passing grade	The student should obtain at least 50% of points at the final test.
laboratory	Passing grade	The student should obtain at least 50% of points from short tests and midterms.

STUDENT'S VOLUME OF WORK

ECTS Balance												
No.	Activity Type	Student Involvement										Unit
		full-time studies					part-time-studies					
		Lec	C	Lab	P	S	Lec	C	Lab	P	S	
1.	Participation in classes according to the schedule	30		30			18		18			h
2.	Other (consultations, exams)	2		2			2		2			h
3.	Total with the direct assist of an academic teacher	64					40					h
4.	Number of ECTS, that student obtains with the direct assist of an academic teacher	2,56					1,6					ECTS
5.	Hours of unassisted student work	61					85					h
6.	Number of ECTS that student obtains working unassisted	2,44					3,4					ECTS
7.	Practical classes volume of work	30					18					h
8.	Number of ECTS obtained by student at practical classes	1,25					0,72					ECTS
9.	Total student's volume of work expressed in hours	125					125					h
10.	ECTS	5										ECTS

BIBLIOGRAPHY

1. Brian W. Kernighan, Denis M. Ritchie, "The C Programming Language", Second Edition, Prentice-Hall Inc., Upper Saddle River, 2012
2. Stephen Prata, "C Primer Plus", 6th Edition, Addison-Wesley, Upper Saddle River, 2015
3. Robert Sedgewick, Kevin Wayne, "Algorithms", 4th edition, Addison-Wesley Inc., Reading, Massachusetts, 2011
4. Jon Bentley, "Programming Pearls" Addison-Wesley, Inc., Upper Saddle River, 2000
5. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "Data Structures and Algorithms", Addison-Wesley Inc., Upper Saddle River, 1987
6. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd edition, MIT Press, Cambridge US, 2009
7. Donald E. Knuth, "The Art of Programming", Vol. 1 -3, Addison-Wesley Inc., Reading, Massachusetts, 1998
8. Steven S. Skiena, "The Algorithm Design Manual", Springer-Verlag, Londyn, 2008