

Subject card

Subject name and code	Computability and Complexity, PG_00203648						
Field of study	Informatics						
Date of commencement of studies	October 2026		Academic year of realisation of subject		2026/2027		
Education level	Master's 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	part-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	academic		Assessment form		credit		
Conducting unit	Institute of Informatics -> Faculty of Mathematics, Physics and Informatics -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		dr Andrzej Borzyszkowski				
	Teachers		dr Andrzej Borzyszkowski				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	20.0	20.0	0.0	0.0	0.0	40
	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	40		0.0		85.0	125
Subject objectives	The aim of this course is to introduce students to the theory of computational complexity, with a particular focus on the concept of computability. Students will learn the concept of a Turing machine, a generic undecidable problem—the halting problem—and the time-complexity classes P, NP, and NP-complete, along with examples of problem reductions.						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[INFMU2_U01] is able to formulate, analyze and solve complex problems related to computer science by selecting and applying appropriate mathematical methods and tools		can identify the complexity class of many problems that arise in computer science can construct an algorithm that proves a problem belongs to a given class		[SU4] test/exam - oral or written		
	[INFMU2_W02] has in-depth knowledge of models of computation, and issues of computational complexity; is familiar with the formal apparatus for formulating and studying the properties of computer objects		is familiar with Turing machines and the resulting existence of undecidable problems; is familiar with the basics of time complexity and the classes P and NP		[SW4] test/exam - oral or written		
	[INFMU2_U03] can design and analyze computational complexity and build algorithms		is able to design and analyze algorithms in terms of computational complexity and implement them		[SU4] test/exam - oral or written		

Subject contents	• Regular and context-free languages • Turing machines, recursively enumerable and recursive languages • The Church–Turing thesis, decidability • Time- and memory-complexity classes. Classes P, NP, NP-complete, polynomial reduction • Computability, universal functions, decidable and undecidable problems, problem reduction, Rice's theorem • Chomsky's hierarchy		
Prerequisites and co-requisites			
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	test	51.0%	40.0%
	test	51.0%	60.0%
Recommended reading	Basic literature	• lecture notes available on the website • J. Jędrzejowicz, A. Szepietowski, Języki, automaty, złożoność obliczeniowa, Wyd. UG, 2008	
	Supplementary literature	• J. E. Hopcroft, J. D. Ullmann, Introduction to Automata, Languages, and Computation, (the more recent edition was co-authored by R. Motwani)	
	eResources addresses		
Example issues/ example questions/ tasks being completed	• We consider the classes PTIME and PSPACE of problems solvable by deterministic Turing machines in polynomial time / in polynomial space. Will these classes change if we allow for non-deterministic Turing machines? • Prove that the halting problem is undecidable. • Given a pushdown automaton, we accept words for which the recognition process ends in an accepting state regardless of the contents of the stack. What class of languages is defined in this way? If we allow the automaton to use two stacks, will the class of languages expands?		
Work placement	Not applicable		

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