

**Subject card**

<b>Subject name and code</b>	Computability and Complexity, PG_00143887						
<b>Field of study</b>	Obliczalność i złożoność (Z)						
<b>Date of commencement of studies</b>	October 2024	<b>Academic year of realisation of subject</b>			2025/2026		
<b>Education level</b>	Master'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>	2	<b>Language of instruction</b>			Polish		
<b>Semester of study</b>	3	<b>ECTS credits</b>			6.0		
<b>Learning profile</b>	academic	<b>Assessment form</b>			exam		
<b>Conducting unit</b>	Institute of Informatics -> Faculty of Mathematics, Physics and Informatics -> Rector						
<b>Name and surname of lecturer (lecturers)</b>	<b>Subject supervisor</b>		dr hab. Christoph Schwarzweller				
	<b>Teachers</b>		dr hab. Christoph Schwarzweller				
<b>Lesson types</b>	<b>Lesson type</b>	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	<b>Number of study hours</b>	20.0	20.0	0.0	0.0	0.0	40
	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>	40		0.0		110.0	150
<b>Subject objectives</b>	The aim of the course is an introduction to computability and the theory of complexity; in particular, that there are undecidable problems and determining whether for a given problem there exists a polynomial algorithm or whether the problem is NP-complete.						

Learning outcomes	Course outcome	Subject outcome	Method of verification
	[INFMU2_K01] knows the limits of his own knowledge and understands the need for further learning	understands the need for further education in the field of computability and complexity	[SK8] obserwacja samodzielnej lub zespołowej pracy studenta
	[INFMU2_U01] can apply mathematical knowledge to formulate, analyze and solve tasks related to computer science	knows Turing machines and the existence of uncomputable problems resulting from them, knows the basics of time complexity and classes P and NP	[SU4] test/egzamin - ustny lub pisemny
	[INFMU2_W01] has in-depth knowledge of the branches of mathematics necessary for the study of computer science; has a good understanding of the role and importance of the construction of mathematical reasoning	knows diagonalization and reduction	[SW4] test/egzamin - ustny lub pisemny
	[INFMU2_W02] has in-depth knowledge of formal languages, models of computation, and issues of computational complexity; is familiar with the formal apparatus for formulating and studying the properties of computer objects	knows Chomsky hierachy	[SW4] test/egzamin - ustny lub pisemny
	[INFMU2_U05] can apply known algorithms in specific situations, can effectively select the type of algorithm depending on the problem at hand	knows the kinds of deterministic, non-deterministic and polynomial algorithms.	[SU4] test/egzamin - ustny lub pisemny
[INFMU2_U04] Defines formal languages using grammars and automata; distinguishes between decidable and undecidable problems.	defines regular, context-free, and context-sensitive languages	[SU4] test/egzamin - ustny lub pisemny	
Subject contents	<p>Turing machines (deterministic and non-deterministic), Turing-Church hypothesis</p> <p>Elements of computability theory (universal functions, decidable and undecidable problems, reduction, Rice's theorem)</p> <p>Formal languages and their decision problems (Chomsky's Hierarchy)</p> <p>Time complexity (DTIME and NTIME classes, Blum's theorem, gap theorem)</p> <p>Classes P and NP</p>		
Prerequisites and co-requisites	Discrete Mathematics		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
		50.0%	100.0%
Recommended reading	Basic literature	Hopcroft, Ullman: Introduction to automata theory, languages, and computation	
		Jędzrejowicz, Szepietwski; Języki, automaty, złożoność obliczeniowa	
	Supplementary literature	Sommerhalder, vanWestrhenen: The theory of computability	
	eResources addresses		
Example issues/ example questions/ tasks being completed			
Work placement	Not applicable		

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