Program	Level	First cycle		
	Name of the program	Theoretical Computer Science, Pure Mathematics, Mathematics Education		
COURSE				
Course title	**Computability**			
Course code	Semester	Course status	ECTS	Contact hours (L+AE+LE)
CS 260	IV	Mandatory course	6	3+2+0
Lecturer				
Course Goals	Introducing students to basic formal computational models.			
Learning Outcomes	A student who successfully completes the course will have the following competencies: • understanding basic terminology from computational theory • understanding the limitations of different computing models • understanding the different types of finite state machines, their formal specifications and properties; • understanding regular expressions and their relationship to finite automata; • ability to design simple deterministic and nondeterministic finite automata; • ability to design simple Turing machines; • understanding the basic classes of complexity.			

COURSE CONTENT

1) Sets, relations, languages; Final representation of language;
2) Finite state machines; Regular expressions; Algorithmic aspects of finite automata;
3) Context-free grammars; Pushdown machines;
4) Definition of a Turing machine; computing with a Turing machine; Direct access Turing machine, Non-deterministic Turing machine;
5) Church-Turing principle;
6) The Halting problem; Unresolved problems with the Turing machine;
7) Decidable and undecidable problems.
8) Chomsky's hierarchy of language.
9) Universal registration machine as a computational model
0) Introduction to computational complexity. Complexity classes: P and NP.

LITERATURE

[1] Hary Lewis, Christos Papadimitriou: Elements of the Theory of Computation, Prentice-Hall, 1997

[2] M. Sipser, Introduction to the Theory of Computation, PWS Publishing Company, 2005.

[3] Michael Garey, David Johnson: Computers and Intractability, A Guide to the Theory of NP-Completness

[4] J. Hromkovic Theoretical Computer Science: Introduction to Automata, Computability, Complexity, Algorithmics, Randomization, Communication, and Cryptography; Springer; 2003;

[5] J. E. Hopcroft, R. Motwani, J. D. Ullman; Introduction to Automata Theory, Languages, and Computation; Addison-Wesley; 2000;

[6] P. Linz, An Introduction to Formal Languages and Automata; Jones & Bartlett Publishers; 2000;

[7] D. C. Kozen, Automata and Computability; Springer; 1997

STUDENT WORKLOAD (hours in a semester)

Lectures	45	Exercises	30	Individual work	75	Total	150

GRADING			**REMARKS**
Criterion	Maximum points	Minimum points	
Midterm exams	45	22	
Projects and Homework assignment	10	5	
Final exam	45	22	
Total	100	55	