DOGUS UNIVERSITY
COME 404 Automata Theory and Formal Languages - Course Instruction Manual

Faculty	Engineering Faculty
Department	Computer Engineering
Credits / Hours	3 (3+0+0)
Course Type	Science Elective
Prerequisites	COME 211
Lecturer	Prof. Dr. Selim AKYOKUŞ (Room: G 912, Tel: (Ext) 1213) Electronic mail: sakyokus@dogus.edu.tr
Textbook	-Michael Sipser, Introduction to the Theory of Computation, 3rd. Ed., 2013.
Recommended Texts	-John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman Title: Introduction to Automata Theory, Languages, and Computation Publisher: Addison-Wesley Year/Edition: 2nd or 3rd edition -Christos Papadimitriou Title: Computational Complexity Publisher: Addison-Wesley Year/Edition: 1994.
Course Description	The objective of this course is to teach some fundamental concepts in automata theory and formal languages including grammar, finite automaton, regular expression, formal language, pushdown automaton, Turing machine and computational complexity. The Properties, capabilities and limitations of each machine/language model, and applications including lexical analysis and parsing will be studied in class.
Purpose	- To understand fundamental concepts in automata theory and formal languages which is the foundation of many branches of computer science. - To teach how to precisely define and reason about computation, and rigorously prove theorems about its capabilities and limitations. - To introduce various models of computation, and is concerned with what can be computed and what cannot be computed by each of these models. - To study computational problems that are computable by a general purpose computer, and investigates how efficiently they can be computed.
Learning Outcomes	The students passing the course will be able to (The letters in parentheses address the relevant program outcomes) I. Be familiar with concepts of abstract machines and formal languages. II. Convert a non-deterministic FA (respectively transition graph) into an equivalent deterministic FA, convert a transition graph or NFA into an equivalent regular expression, and convert a regular expression into an equivalent FA. Construct a regular expression (respectively a context-free grammar) for a regular language (1a, 1b, 2,3) III. Convert a context-free grammar into an equivalent pushdown automaton. Construct a context-free grammar for a given context-free language. (1a, 1b, 2,3) IV. Build simple Turing machines. . Compare FA, PDA, and Turing machines. (1a, 1b, 2,3) V. Apply a parsing algorithm. Build a parse tree or a derivation from a context-free grammar. Compare regular, context-free, recursive, and recursively enumerable languages. (1a, 1b, 2,3) VI. Be familiar with the implications of Church-Turing thesis. Understand that there are problems for which an algorithm exists, and problems for which there are no algorithms (non-recursive, non-recursively enumerable languages) and understand the implications of such results. Understand and explain the diagonalization process as used in proofs about computability. Class P, Class NP, NP-complete problems (1a, 1b, 2,3)
Content of the Course	Regular languages, finite automata, non-deterministic FA, Context-free languages, pushdown automata, Parsing, normal forms, ambiguity, Pumping lemmas and closure properties, Turing machines and other equivalent models, Decidable languages, non-decidable languages, recognizable languages, Chomsky hierarchy, Computational complexity.
General Skills	Acquire theoretical skills

Learning Methods	Formal lectures for conceptual knowledge; exercises and home works to practice concepts presented in lectures.
Assessment	Homeworks (10%) Midterm 1 (23%) Midterm 2 (23%) Final (44%)

Course Plan

Week	Topics	Learning Outcomes
1	Introduction, mathematical preliminaries, Proof techniques. Alphabets, language, regular expression.	I
2	Deterministic finite automata, Nondeterministic finite automata	II
3	Regular languages and their closure properties	II
4	Regular expressions. Pumping lemma for regular languages	II
5	Context-free grammars	III,V
6	Context-free languages (CFL) and their closure properties, Pumping lemma for CFLs	III
7	Pushdown automata (PDAs): deterministic and nondeterministic.	III
8	Turing machines (TMs), their instantaneous descriptions. Language acceptance by TMs.	III
9	Turing machine variants and the Church-Turing thesis. Chomsky hierarchy	IV
10	Decidable languages	IV
11	Diagonalization and the halting problem	VI
12	Undecidable languages	VI
13	Computational Complexity	VI
14	Class P, Class NP, NP-complete problems	VI

Relationship Between the Course and Computer Engineering Program Outcomes	
Program Outcomes	**Contribution Level**
1a. Sufficient knowledge on mathematics, science, and computer engineering.	◒
1b. Ability to apply theoretical and practical knowledge of these fields for modelling and solving computer engineering problems.	●
2. Ability to identify, define, formulate, and solve complex (open ended) computer engineering problems; ability to select and apply the appropriate modelling and analysis methodology for this purpose	◒
3. Ability to design a complex system, process, device or product; ability to apply modern design methods for this purpose.	◒
4. Ability to develop, choose, and use modern methods and tools that are required for computer engineering applications	
5. Ability to design and conduct experiments as well as collect data, analyze and interpret results for examining computer engineering problems	
6. Ability to function on interdisciplinary and multidisciplinary teams; ability to function on individually	
7. Ability to communicate effectively with written, oral and visual forms.	
8. Recognition of the need for lifelong learning, and the ability to access knowledge, to pursue the developments in science and technology and to remain up to date.	
9. Consciousness of professional social and ethical responsibility.	
10a. Knowledge about professional applications such as project, risk, and change management.	
10b. Awareness of innovativeness, entrepreneurship and sustainable development.	
11a. A diverse background providing the ability to discuss and evaluate the impact of engineering solutions on health, environment and safety in global and societal context.	
11b. Up to date with contemporary issues	
11c. Awareness of legal consequences of engineering solutions	
● ***Major Contribution (3),*** ◒ ***Moderate Contribution (2),*** ◒ ***Minor Contribution (1)***	