ADA University, School of IT and Engineering (SITE)

Syllabus of the courses MATH 1101 Discrete Structures and Math 1102 Fundamentals of Mathematics

(*6 ECTS,* Fall 2023**)**

Syllabus is designed according to the recommendations of Curriculum Guidelines for Undergraduate Degree Programs in Computer Science, developed by ACM and IEEE.

Course Information

Math 1101 Discrete Structures (CRNs 10381, 10382, 10383)

Fuad Hajiyev, Ph.D. in Math., Assistant Professor School of IT and Engineering, ADA University, Baku, Azerbaijan
Joint Lectures: CRNs 10381, 10382, 10383
Problem solving session: CRNs 10381
Office: Building B (Campus), room 222
Tel: +99412 4373235 (ext 415)
Email: fhajiyev@ada.edu.az , www.ada.edu.az

Samin Malik, Ph.D. in Math., Assistant Professor School of IT and Engineering, ADA University, Baku, Azerbaijan
Problem solving sessions: CRNs 10382, 10383
Office: Building B (Campus), room 219
Tel: +99412 4373235 (ext 488)
Email: smalik@ada.edu.az , www.ada.edu.az

Teaching Hours:

Joint Lectures – Instructor Dr. Fuad Hajiyev:

Attendance: **Two options are available for students:**

- **On-line mode, LMS Blackboard (BB);**
- **On-campus mode is also available with a limited number of seats: (first come - first served).**

Tue, 08:30 – 09:45 (venue: D207 for on-campus mode with **max 82 seats**)
https://eu.bbcollab.com/guest/809512785188403ab482194514ceb2b5 (guest link if needed).

Thur, 08:30 – 09:45 (venue: D209 for on-campus mode with **max 39 seats**)
https://eu.bbcollab.com/guest/809512785188403ab482194514ceb2b5 (guest link if needed).

It is highly recommended to use the regular link from the student's personal account in the BB.

Problem solving session (PSS) – Instructor Dr. Fuad Hajiyev

CRN 10381, Fri: 10:00 – 11:15; Venue D108. Guest link for students from other CRNs - https://eu.bbcollab.com/guest/6db4c97e059d4a59ac838d5361be32fd

Problem solving sessions (PSS) – Instructor Dr. Samin Malik

CRN 10382, Fri: 11:30 – 12:45; Venue D108. Guest link for students from other CRNs - TBA

CRN 10383, Fri: 13:00 – 14:15; Venue D108. Guest link for students from other CRNs - TBA

Course materials are available through BB as following:

1. All students, regardless of which CRN they are registered with, have an equal access to all PSS's and can attend classes with the same topic of discussion multiple times **by visiting classes in other CRNs (online or on-campus)**.
2. **Textbooks** are placed in the folder ⇒: Course Content/Textbooks
3. **Lecture Notes** are available in the folder ⇒ Course Content/Lecture Notes
4. **Homework Assignments** will be placed in the folder ⇒ Course Content /Homework Assignments
5. **Quizzes, Midterm and Final Exams tasks** will be published in ⇒ Course Content/ Exams and Quizzes
6. **Syllabus** is available in the folder ⇒ Course Content/Syllabus

Introduction.

Discrete Structures (DS) or Fundamental of Mathematics (FM) is a mandatory component of Math, IT-oriented, and Engineering Bachelor programs in School of IT and Engineering at ADA University.

DS (FM) is one of the important parts of the mathematical background required by mathematicians, engineers, computer scientists, ICT specialists, physicists, economists, statisticians.

In university curricula, you can find other names for the proposed course, for example: "Discrete Mathematics", "Finite Mathematics", "Fundamental Structures of Computer Science" (for computer science students) or "Mathematical Foundations of Computer Science".

Where does the course fit in your curriculum?

This course covers all the required Discrete Structures (Discrete Mathematics) for Math, Computer, Information Technologies, and Engineering majors.

Course textbooks:

Main

[1]. Kenneth Rosen: “Discrete Mathematics and Its Applications”, 7th edition, 2012, McGraw-Hill.

[2]. Seymour Lipschutz, Marc Lipson: Discrete Mathematics. (Shaum’s Outline), 4th edition, 2009, McGraw-Hill.

Supplementary

[3]. Eric Lehman, Google Inc; F Thomson Leighton, Dept of Mathematics & CS, and CS and AI Laboratory, MIT; Albert R Meyer, Dept of EE & CS, and CS and AI Laboratory, MIT: “Mathematics for Computer Science” (revised 6th June, 2018). - https://courses.csail.mit.edu/6.042/spring18/mcs.pdf

Knowledge units and topics covered in the course

Knowledge Area	**Hours of Coverage (Theoretical Part – Lectures)**	**Problem solving session (recitation)**
Discrete Structures (DS)	37.5h=(2.5h/w)*(15 weeks)	18.75 h=1.25h/w*15 weeks

Body of Knowledge coverage

KA	**Knowledge Unit**	**Topics Covered**	Hours	Week #	**Lecture Notes #**	Textbooks
DS	Basic Discrete Structure: Algebra of Sets.	Introduction to the course. Sets and Functions. Set Operations. Algebra of sets. Cardinality of sets.	2.5 h	1	1-2	[1]. p. 115-177 [2]. p. 1-11, 12-22, 43-55, 60-69
DS	Algebras of Propositions and Predicates.	Propositions. Logical Operations. Algebra of propositions. Propositional functions (Predicates) and Quantifiers. Algebra of Predicates.	5 h	2-3	3-6	[1]. p. 1-69 [2]. p. 70-87
DS	Algebra of Circuits. Boolean Algebras.	Logic Gates and Circuits. Algebra of Circuits. Boolean Algebras. Sum-of-Product Forms (SoP); Complete Sum-of-Product Forms (CSoP) Prime Implicants; Minimal Boolean Expressions (MSoP). Karnaugh Maps Method to find MSoP.	5 h	4-5	7-10	[1]. p. 811-846, Ch. 12 [2]. p. 368-408, Ch. 15
DS	Proof Technique I: Mathematical Induction.	Principle of Mathematical Induction I. Principle of Mathematical Induction II (Strong Induction).	1.25 h	6	11	[1]. p. 311-344, Ch 5.1-5.2 [2]. p. 12, Ch 1; 266-267, 284-285, Ch. 11

DS	Proof Technique II: Recursive Definitions, Structural Induction.	Recursively Defined Functions. Recursively Defined Sets and Structures. Structural Induction.	1.25 h	6	12	[1]. p. 340-344, Ch 5.2, p.345-360, Ch 5.3;
DS	Finite Sets, Counting Principles, Elements of Combinatorics	Sum Rule Principle. Product Rule Principle. Inclusion–Exclusion Principle. Pigeonhole Principle. Factorial Function and Binomial Coefficients, Permutations and Combinations	3.75 h	7-8	13-15	[1]. p. 385-444, Ch 6 [2]. p. 88-111, 118-122, Ch 5-6
DS	Discrete Probability	Introduction to Discrete Probability. Finite Probability. Probability Theory. Assigning Probabilities. The Probabilistic Method. Bayes' Theorem. Bayesian Spam Filters. Expected Value and Variance. Expected Values. Linearity of Expectations. Average-Case Computational Complexity. The Geometric Distribution. Independent Random Variables. Variance. Chebyshev's Inequality.	6.25 h	8-10	16-20	[1]. p. 445-495, Ch 7 [2]. p. 123-149, Ch 7
DS	Graphs and Trees	Graphs and Graph Models, Representing Graphs. Connected Graphs, Eulerian and Hamiltonian Graphs. Shortest Paths in weighted Graphs (Dijkstra Algorithm). Trees. Spanning Tree Algorithms.	3.75 h	11-12	22-24	[1]. p. 641-717, Ch 10 p. 745-803, Ch 11
DS	Relations	Relations and their properties. Operations over relations. Function as a particular Case of a Relation. Equivalence Relation.	1.25	11	21	[1]. p. 573-640, Ch 9 [2]. p. 23-42, Ch. 2
DS	Algorithms and Complexity	Intro to the notions of algorithm and complexity. Growth of functions: Big-O, Big-Ω, and Big Θ notations. Searching and Sorting algorithms and complexity issues (additional reading).	2.5 h	13	25-26	[1]. p. 191-236, Ch 3 [2]. p. 57-60, Ch. 3,
DS	Languages and Automata (Modeling of Computations)	Languages and Regular Expressions. Deterministic Finite State Automata (DFSA) and Language Recognition Problem. Nondeterministic Finite State Automata (NDFSA) and Kleene's Theorem. Pumping Lemma. Finite-State Machines with Outputs.	5 h	14-15	27-30	[1]. p. 847-900, Ch 13 [2]. p. 303-323, Ch. 12, p. 323-336,Ch 13

Why do you teach the course this way (Program Objectives)?

Discrete Structures is a transfer-oriented course designed to meet the requirements of Computer and Information Science degree programs.

Furthermore, this course is designed to meet the following program objectives. Upon successful completion of this course, students will be able to:

- Demonstrate critical thinking, analytical reasoning, and problem-solving skills.
- Apply appropriate mathematical concepts and operations to interpret data and to solve problems.
- Identify a problem and analyze it in terms of its significant parts and the information needed to solve it.
- Formulate and evaluate possible solutions to problems and select and defend the chosen solutions.

Learning outcomes:

1. Explain with examples the basic terminology of functions, relations, and sets.
2. Perform the operations associated with sets, functions, and relations.
3. Convert logical statements from informal language to propositional and predicate logic expressions.
4. Apply formal methods of symbolic propositional and predicate logic.
5. Use the rules of inference to construct proofs in propositional and predicate logic.
6. Describe the strengths and limitations of propositional and predicate logic.
7. Outline the basic structure of each proof technique (direct proof, proof by contradiction, and induction) described in this unit.
8. Apply proof techniques (direct proof, proof by contradiction, and induction) correctly in the construction of a sound argument.
9. Determine which type of proof is best for a given problem.
10. Explain the parallels between ideas of mathematical and/or structural induction to recursion and recursively defined structures.
11. Explain the relationship between weak and strong induction and give examples of the appropriate use of each.
12. State the well-ordering principle and its relationship to mathematical induction.
13. Apply counting arguments, including the sum and product rules, the inclusion-exclusion principle, and the arithmetic/geometric progressions.
14. Apply the pigeonhole principle in the context of a formal proof.
15. Compute permutations and combinations of finite sets.
16. Apply the Boolean Algebra Rules and Logic Gates to Digital Circuits Design.
17. Be familiar with elementary concepts of Languages and Automata theory and understand their role in Computer Science.

What is the format of the course?

The material is presented in 75 minutes 2 lectures in a week (by 75 min each) in a week (2.5 h/w), during the 15 weeks, with problem-solving (recitation) sessions, 75 min in a week (1.25 h/w).

How are students assessed (requirements and the breakdown of the grading)?

Total Grade (TG)=100 points and consists of the following components:

1. Homework Assignments: 3pt*5=15 pt.
2. **Midterm 1 Exam – 25 pt (week 5), Midterm 2 – 20 pt (week 9), and Midterm 3 – (week 14) – 20 pt**

3. **Final Exam: 20 pt**.

Only those with documented medical or other emergencies may make-up a missed exam.

Honesty: There is zero tolerance for any kind of cheating.

Important: As you are the central figure in the teaching-learning process your effort will be decisive in achieving the goals of this course. Keep in mind that a lecturer is one who only helps you to achieve these goals.

Reading textbooks is of utmost importance. Without this type of work, it is very unlikely that you will succeed in this course.

Grading Procedures: Students will be graded on absolute scale provided below.

A+	A	A-	B+	B	B-	C+	C	C-	D+	D	F
≥98	≥93.5	≥89.5	≥86.5	≥82.5	≥79.5	≥76.5	≥72.5	≥69.5	≥66.5	≥59.5	<59.5

Passing Grade for the Discrete Structures course is “D” (59.5).

Exam Rules. A missed examination counts as a zero.

Attendance Policy. The regular attendance is highly recommended.

Note. Instructors could modify schedule and content of the classes, if necessary.

Good Luck!

Dr. Fuad Hajiyev