COURSE DESCRIPTION FORM

INSTITUTION National University of Computer and Emerging Sciences (NUCES-FAST)

PROGRAM (S) TO BE EVALUATED BS(AI)

A. Course Description

Course Code	CS-211
Course Title	Discrete Structures
Credit Hours	3+0
Prerequisites by Course(s) and Topics	None
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	Midterm examination: 15% Midterm examination: 15% Assignments/Quizzes: 20% End-term examination 50%
Course Coordinator	**Mr. Shoaib Raza**
URL (if any)	https://classroom.google.com/u/0/c/MjYzNTE4MDMwODUx
Current Catalog Description	Logic, relations, functions, basic set theory, counting, proof techniques, mathematical induction, graph theory, recursion, recurrence relations, number theory and sequence & series. All the topics will be taught in perspective of their applications in computing.
Textbook (or **Laboratory Manual** for Laboratory Courses)	Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw Hill, 8th Edition, 2019.
Reference Material	Sussana S. Epp, Discrete Mathematics with Applications, Brooks Cole, Cengage Learning, 5th Edition, 2020.
Course Goals	A discrete mathematics course has more than one purpose. Students should learn a particular set of mathematical facts and how to apply them; more importantly, such a course should teach students how to think logically and mathematically. To achieve these goals, the focus of this course is on basic mathematical concepts in discrete mathematics and on applications of discrete mathematics in algorithms and data structures. The focus is also on

	teaching the problem-solving strategies, techniques, and tools and to show students how discrete mathematics can be used in modern computer science. In particular, this course is meant to introduce logic, proofs, sets, relations, functions, counting, and probability, with an emphasis on applications in Computer Science. Further, this course aims to develop understanding and appreciation of the finite nature inherent in most Computer Science problems and structures through study of combinatorial reasoning, abstract algebra, iterative procedures, predicate calculus, tree and graph structures.

A. Course Learning Outcomes (CLOs)

Course Learning Outcomes (CLOs):		
At the end of the course the students will be able to:	**Domain**	**BT Level***
1. **Understand** the key concepts of Discrete Structures such as Sets, Permutations, Relations, Graphs, and Trees etc.	C	2
2. **Apply** formal logic proofs and/or informal, but rigorous, logical reasoning to real problems, such as predicting the behavior of software or solving problems such as puzzles.	C	3
3. **Apply** discrete structures into other computing problems such as formal specification, verification, databases, artificial intelligence, and cryptography.	C	3
4. **Differentiate** various discrete structures and their relevance within the context of computer science, in the areas of data structures and algorithms, in particular.	C	4
* BT= Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A= Affective domain		

B. Program Learning Outcomes

For each attribute below, indicate whether this attribute is covered in this course or not. Leave the cell blank if the enablement is little or non-existent.

1. Academic Education:	To prepare graduates as computing professionals	✔
2. Knowledge for Solving Computing Problems:	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.	✔
3. Problem Analysis:	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.	✔
4. Design/ Development of Solutions:	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	✔
5. Modern Tool Usage:	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.	✔

B. Program Learning Outcomes

For each attribute below, indicate whether this attribute is covered in this course or not. Leave the cell blank if the enablement is little or non-existent.

6. Individual and Team Work:	Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings.	✔
7. Communication:	Communicate effectively with the computing community and with society at large about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.	✔
8. Computing Professionalism and Society:	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.	
9. Ethics:	Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.	✔
10. Life-long Learning:	Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.	✔

Topics Covered in the Course, with Number of Lectures on Each Topic (assume 15-week instruction and one-hour lectures)

1. Topics to be covered:

List of Topics	No. of Weeks	Contact Hours	CLO
Chapter 1: The Foundations: Logic and Proofs Introduction Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference **Chapter 2: Sets, Functions, Sequences and Sums** Sets, Set Operations, Functions, Sequences and Series **Chapter 9: Relations** Relations and their Properties, Applications of Relations, Representing Relations, Equivalence Relations, and Partial Orderings	**5**	**15**	**1,2,3**
========== ***MID 1*** ==========			
Chapter 4: Number Theory and Cryptography Divisibility and Modular Arithmetic, Integer Representation and Algorithms, Primes and Greatest Common Divisors, Congruences and Applications and Cryptography **Chapter 5: Induction and Recursion** Mathematical Induction and Recursive Algorithms **Chapter 6 and 8: Counting & Counting Techniques** Basics, Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Recurrence Relations	**5**	**15**	**1,2, 3**
========== ***MID 2*** ==========			

	Chapter 10 Graphs Graphs and Graph Models, Terminologies, Types of Graphs, Representing Graphs and Isomorphism, Connectivity, Euler and Hamiltonian Paths, Planar Graphs, and Graph Coloring **Chapter 11 Trees** Introduction, Applications, Tree Traversal, Spanning Trees and Minimum Spanning Trees **Chapter 1** Introduction to Proofs and Proof Methods **Chapter 3 Algorithms** Algorithms, The Growth of Functions, and Complexity of Algorithms	**4.5**	**14**	**1,2, 3,4**
	========== ***END-TERM (FINAL) EXAM*** ==========			
	Review	**0.5**	**1**	**1,2,3,4**
	Total	**15**	**45**	

Laboratory Projects/Experiments Done in the Course	No Labs and Projects in this course.			
Programming Assignments Done in the Course	Some optional programming assignments will be given.			
Class Time Spent on (in contact hours)	**Theory**	**Problem Analysis**	**Solution Design**	**Social and Ethical Issues**
	10	15	20	0
Oral and Written Communications	Students need to participate in class discussion and class assignments.			

Instructor Name **Mr. Shoaib Raza**

Instructor Signature _____

Date: **February 7, 2021**