NEW YORK CITY COLLEGE OF TECHNOLOGY
The City University of New York

DEPARTMENT:	Mathematics
COURSE:	MAT 2440
TITLE:	Discrete Structures and Algorithms I
DESCRIPTION:	This course introduces the foundations of discrete mathematics as they apply to computer science, focusing on providing a solid theoretical foundation for further work. Topics include functions, relations, sets, simple proof techniques, Boolean algebra, propositional logic, elementary number theory, writing, analyzing and testing algorithms.
TEXT:	Discrete Mathematics and Its Applications,8th edition by Kenneth H. Rosen McGraw-Hill
CREDITS:	3 (2 class hours, 2 lab hours)
PREREQUISITES:	MAT 1375 or higher and one of the following: CST 1201 or CST 2403 or MAT 1630

Prepared by Professors Henry Africk, Brad Isaacson, Caner Koca, Nan Li, Satyanand Singh, Arnavaz Taraporevala, Johann Thiel. (Fall 2017)
Revised by Professors Arnavaz Taraporevala, Johann Thiel (Spring 2020)
Revised (Spring 2022)

A. Testing Guidelines:
   The following exams should be scheduled:
   1. A one-hour exam at the end of the First Quarter
   2. A one-session exam at the end of the Second Quarter
   3. A one-hour exam at the end of the Third Quarter
   4. A one-session Final Examination

B. A Computer Algebra System will be used in class and for a project.

Course Learning Outcomes	General Education Learning Outcomes	Flexible Core- Scientific World
Use the rules of logic to understand mathematical statements and prove propositions using: • A direct proof • An indirect proof • A proof by contradiction • A proof by induction	Acquire strategies and skills to create and analyze mathematical proofs. Be able to determine the implications and consequences of a given set of mathematical statements. Be able to use appropriate language to communicate mathematical ideas.	Evaluate evidence and arguments critically or analytically. Produce well-reasoned written or oral arguments using evidence to support conclusions.
Learn and understand basic notation, definitions, and concepts related to sets, functions, and sequences.	Acquire a foundation of knowledge of important mathematical concepts and definitions.	Identify and apply the fundamental concepts and methods of a discipline or interdisciplinary field exploring the scientific world, including, but not limited to: computer science, logic, and mathematics.
Write simple algorithms using pseudocode and understand the efficiency of algorithms.	Be able to understand the limitations and implications of an algorithm. Be able to analyze pseudocode to determine the functionality and efficiency of an algorithm. Be able to generate algorithms and effectively communicate their purpose.	Gather, interpret, and assess information from a variety of sources and points of view. Demonstrate how tools of science, mathematics, technology, or formal analysis can be used to analyze problems and develop solutions.
Understand basic number theory topics.	Be able to solve problems using number-theoretic techniques. Be able to understand the implications of number-theoretic results with respect to cryptography.	Identify and apply the fundamental concepts and methods of a discipline or interdisciplinary field exploring the scientific world, including, but not limited to: computer science, logic, and mathematics. Understand the scientific principles underlying matters of policy or public concern in which science plays a role.

New York City College of Technology Policy on Academic Integrity

Students and all others who work with information, ideas, texts, images, music, inventions, and other intellectual property owe their audience and sources accuracy and honesty in using, crediting, and citing sources. As a community of intellectual and professional workers, the College recognizes its responsibility for providing instruction in information literacy and academic integrity, offering models of good practice, and responding vigilantly and appropriately to infractions of academic integrity. Accordingly, academic dishonesty is prohibited in The City University of New York and at New York City College of Technology and is punishable by penalties, including failing grades, suspension, and expulsion. The complete text of the College policy on Academic Integrity may be found in the catalog.

Writing Intensive Course Designation

This course has been designated as a "Writing Intensive" (WI) course by City Tech. A WI course includes critical reading, logical thinking, and the use of writing to help students understand the topic; the use of appropriate style and disciplinary conventions in writing and speaking; the use of research resources, including the library, specific to the discipline; a detailed syllabus; a comprehensive course calendar; and a minimum of fifteen pages of writing per student. Writing assignments will be both formal (graded) and informal (non-graded).

Written work will be a mandatory part of the course. This can include coding projects, proofs, and written assignments. Written work will account for a minimum of 10% of the overall grade in the course.

MAT 2440 Discrete Structures and Algorithms I
Text: Discrete Mathematics and its Applications,8th edition, by Rosen

Lec.	Discrete Structures and Algorithms I	Homework
1	1.1 Propositional Logic (1-13)	(P. 13) 1, 3, 11, 13, 25, 29, 31, 33, 39, 48
2	1.2 Applications of Propositional Logic (17-23)	(P. 23) 1-3, 7, 44, 45
3-4	1.3 Propositional Equivalences (26-37)	(P. 38) 3, 4, 6, 7, 8, 11, 27, 46*
5	1.4 Predicates and Quantifiers (40-56) 1.5 Nested Quantifiers (60-68)	(P. 56) 1, 2, 4, 7-13 odd, 19, 30, 38 (P. 68) 1, 3, 10, 27, 31, 33
6	1.6 Rules of Inference (73-82)	(P. 82) 5, 6, 19, 20, 35
7	1.7 Introduction to Proofs (84-95)	(P. 95) 1-4, 9-12, 19, 20
8	**Test 1**	
9	2.1 Sets (121-131)	(P. 131) 1, 7, 13, 21 (a)&(b), 29, 30, 33, 37
10	2.2 Set Operations (133-144)	(P. 136) 3, 15(b), 19(b), 23-25, 28, 52, 53, 58-60
11	2.3 Functions (147-161)	(P. 161) 3, 9-12, 15, 20, 23, 30, 33, 41, 44-46, 60-63
12	2.4 Sequences and Summations (163-177)	(P. 177) 3, 9, 25, 29-31, 33, 35, 45
13-15	3.1 Algorithms (201-213)	(P. 213) 1, 3-15 odd, 16-18, 36, 37, 39-41, 56, 57
16	**Test 2**	
17-18	3.2 The Growth of Functions (216-228)	(P. 228) 1-27 odd, 34-42
19-21	3.3 Complexity of Algorithms (231-241)	(P. 241) 1-5, 20, 22, 40
22	4.1 Divisibility and Modular Arithmetic (251-258)	(P. 258) 1, 13, 21, 27, 31-35, 36, 47
23	4.2 Integer Representations and Algorithms (260-268) 4.3 Primes and Greatest Common Divisors (271-288)	(P. 269) 1-15 odd, 57, 58 (P. 288) 1, 3, 17, 25, 27, 33
24	4.4 Solving Congruences (290-300) 4.5 Applications of Congruences (303-308)	(P. 301) 1, 5, 11, 21, 33, 34, 55 (P. 308) 1-5 odd
25	4.6 Cryptography (310-321)	(P. 322) 1-5 odd
26	**Test 3**	
27-28	5.1 Mathematical Induction (331-350)	(P. 350) 1-11 odd, 15, 21
29	**Review**	
30	**Final Exam**	

MAT 2440 Discrete Structures and Algorithms I
Text: Discrete Mathematics and its Applications,7th edition, by Rosen

Lec.	Discrete Structures and Algorithms I	Homework
1	1.1 Propositional Logic (1-12)	(P. 12) 1, 3, 9, 11, 23, 27, 29, 31, 37, 44
2	1.2 Applications of Propositional Logic (16-22)	(P. 22) 1-3, 7, 40, 41
3-4	1.3 Propositional Equivalences (25-34)	(P. 34) 3, 4, 6, 7, 8, 9, 23, 42*
5	1.4 Predicates and Quantifiers (36-52) 1.5 Nested Quantifiers (57-64)	(P. 53) 1, 2, 4, 7-13 odd, 19, 30, 36 (P. 64) 1, 3, 10, 27, 31, 33
6	1.6 Rules of Inference (69-78)	(P. 79) 5, 6, 19, 20, 35
7	1.7 Introduction to Proofs (80-90)	(P. 91) 1-4, 9-12, 17, 18
8	**Test 1**	
9	2.1 Sets (115-125)	(P. 125) 1, 5, 11, 19(a)&(b), 27, 28, 31, 35
10	2.2 Set Operations (127-135)	(P. 136) 3, 15(b), 17(b), 21-23, 26, 46, 47, 52-55
11	2.3 Functions (138-152)	(P. 152) 3, 9, 10-12, 15, 20, 23, 30, 33, 39, 42-44, 58-61
12	2.4 Sequences and Summations (156-167)	(P. 167) 3, 9, 25, 29, 30, 31, 33, 35, 43
13-15	3.1 Algorithms (191-202)	(P. 202) 1, 3-15 odd, 16-18, 34, 35, 37-39, 52, 53
16	**Test 2**	
17-18	3.2 The Growth of Functions (204-216)	(P. 216) 1-27 odd, 34-42
19-21	3.3 Complexity of Algorithms (218-229)	(P. 229) 1-5, 20, 22, 36
22	4.1 Divisibility and Modular Arithmetic (23-244)	(P. 244) 1, 9, 15, 21, 26-29, 30, 21
23	4.2 Integer Representations and Algorithms (245-254) 4.3 Primes and Greatest Common Divisors (257-272)	(P. 255) 1-15 odd, 51, 52 (P. 272) 1, 3, 17, 25, 27, 33
24	4.4 Solving Congruences (274-284) 4.5 Applications of Congruences (287-292)	(P. 284) 1, 5, 11, 21, 33, 34, 55 (P. 292) 1-5 odd
25	4.6 Cryptography (294-303)	(P. 304) 1-5 odd
26	**Test 3**	
27-28	5.1 Mathematical Induction (311-329)	(P. 329) 1-11 odd, 15, 21
29	**Review**	
30	**Final Exam**	

List of Suggested Projects

1. Lecture 13 & 14: Implement the max and linear search algorithms in a programming language.
2. Lecture 21: Timing algorithms by input size.
3. Lecture 23: Primality testing using a programming language.
4. Lecture 24: Implement a hashing function and a pseudorandom generator in a programming language.
5. Lecture 25: Implement a Caesar cipher.
6. Lecture 27: Implement a Tower of Hanoi game.