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,7th edition by Kenneth H. Rosen
McGraw-Hill

CREDITS: 3 (2 class hours, 2 lab hours)

PREREQUISITES: MAT 1375 of higher, and CST 2403 or CST 1201

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

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 Intended Learning Outcomes/Assessment Methods

Learning Outcomes	**Assessment Methods**
1. 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	Classroom activities and discussion, homework, project, exams.
2. Write simple algorithms using pseudocode and understand the efficiency of algorithms.	Classroom activities and discussion, homework, project, exams.
3. Understand basic number theory topics.	Classroom activities and discussion, homework, project, exams.
4. Use computer technology to assist in the above.	Classroom activities and discussion, homework, project, exams.

General Education Learning Outcomes/Assessment Methods

Learning Outcomes	**Assessment Methods**
1. Gather, interpret, evaluate, and apply information discerningly from a variety of sources.	Classroom activities and discussion, homework, project, exams.
2. Understand and employ both quantitative and qualitative analysis to solve problems.	Classroom activities and discussion, homework, project, exams.
3. Employ scientific reasoning and logical thinking.	Classroom activities and discussion, homework, project, exams.
4. Communicate effectively using written and oral means.	Classroom activities and discussion, homework, project, exams.
5. Utilize computer based technology in accessing information, solving problems and communicating.	Classroom activities and discussion, homework, project, exams.
6. Work with teams. Build consensus and use creativity.	Classroom activities and discussion, project, homework.
7. Acquire tools for lifelong learning.	Classroom activities and discussion, homework, project, exams.

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.

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	1.3 Propositional Equivalences (25-34)	(P. 34) 3, 4, 6, 7, 8, 9, 23
4	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
5	1.6 Rules of Inference (69-78)	(P. 79) 5, 6, 19, 20, 35
6	1.7 Introduction to Proofs (80-90)	(P. 91) 1-4, 9-12, 17, 18
7	**Test 1**	
8	2.1 Sets (115-125)	(P. 125) 1, 5, 11, 19(a)&(b), 27, 28, 31, 35
9	2.2 Set Operations (127-135)	(P. 136) 3, 15(b), 17(b), 21-23, 26, 46, 47, 52-55
10	2.3 Functions (138-152)	(P. 152) 3, 9, 10-12, 15, 20, 23, 30, 33, 39, 42-44, 58-61
11	2.4 Sequences and Summations (156-167)	(P. 167) 3, 9, 25, 29, 30, 31, 33, 35, 43
12	2.5 Cardinality of Sets (170-176) (optional)	(P. 176) 1, 3, 11
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
28	5.5 Program Correctness (372-376) (optional)	(P. 377) 3,7
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.