CS 2300 Course Syllabus

For additional course information, including prerequisites, corequisites, and course fees, please refer to the Catalog: https://catalog.uvu.edu/

Semester: Spring
Year: 2025
Course Prefix: CS
Course and Section #: CS 2300 – X01
Course Title: Discrete Mathematical Structures 1
Credits: 3

Course Description

For Computer Science Majors.
Covers algebraic structures applied to computer programming. Includes logic, sets, elementary number theory, mathematical induction, recursion, algorithm complexity, combinatorics, and relations. First of a two-semester sequence.
This course is part of the Computer Science discipline core requirements and must be successfully completed to fulfill graduation requirements for the Computer Science degree. (You must earn a grade C+ or higher to be matriculated in the CS programs.)

Course Attributes

This course has the following attributes:
☐ General Education Requirements
☐ Global/Intercultural Graduation Requirements
☐ Writing Enriched Graduation Requirements
☒ Discipline Core Requirements in Program
☐ Elective Core Requirements in Program
☐ Open Elective
Other: *Click here to enter text.*

Instructor Information

Instructor Name: Robert Kumar

Student Learning Outcomes

Upon successful completion, students should be able to:
- Use symbolic logic to determine proper conditional clauses in a program
- Prove theorems algebraically
- Perform basic operations on sets
- Determine the nature of the growth of a function or algorithm
- Prove properties of recursive algorithms using mathematical induction
- Use relations and functions in the design of algorithms and applications

Course Materials and Texts

Click any zyBooks assignment link in the learning management system (Do not go to the zyBooks website and create a new account). Click Subscribe

Course Requirements

Course Assignments, Assessments, and Grading Policy

Assessments:

There are 4 exams (3 midterms exam and a final exam). All exams are multiple-choice and are administered in Canvas over a 3-day period using Proctorio. Students must be alone when taking exams. Exams are timed and must be completed before 11:59pm on the last of the three exam days. The final exam is comprehensive.

Due Dates:

Students are expected to submit assignments before the due date as listed in the LMS. Late assignments submission will incur a late penalty.

Grading Scale:

The following grading standards will be used in this class:

Grade	Percent
A	94-100
A-	90-93
B+	87-89
B	83-86
B-	80-82
C+	77-79
C	73-76
C-	70-72
D+	67-69
D	63-66
D-	60-62
E	0-59

Late Work Statement:

All work is due at 11:59pm on its respective due date. Late work is automatically penalized 10% per day by Canvas, so there can be no exceptions for late penalties. Plan on getting your work done ahead of time. NO WORK WILL BE ACCEPTED AFTER THE LAST DAY OF CLASS UNLESS AUTHORIZED BY THE SCHOOL.

Required or Recommended Reading Assignments

This course is organized into 14 modules, with reading and assignments as follows.

Module 1: Logic, Proposition, Conditional Statement, Logical Equivalence, Laws of Propositional Logic

Module 2: Quantifier Statements, Nested Quantifiers, Translating with Nested Quantifiers, Predicates and Quantifiers

Module 3: Forming an Argument, Rules of Inference, Inference Rules for Quantified Statements, Logical Paradox

Module 4: Parity, Direct Proof, Disprove by a Counterexample, Proof by Contrapositive, Proof by Contradiction, Proof by Cases,

Module 5: Introduction to Sets, Union, Intersection, and Difference of Sets, Symmetric Difference, Set Identities, Cartesian Product, Partition of a Set

Module 6: Introduction to Functions, Floor and Ceiling Functions, Function Properties, Inverse Function and Composition of Functions, Asymptotic Growth of Functions

Module 7: Introduction to Relations, Properties of Binary Graphs, Composite Relation, Graph Power, and Transitive Closure, Equivalent Relations,

Module 8: Division Algorithm, Modular Arithmetic, Prime and Prime Factorization, GCD and LCM, Euclidean Algorithm, Number Conversions, Fast Exponentiation, Introduction to Cryptography

Module 9: Sequences, Summation of Sequences, Recurrence Relations, Mathematical Induction, Strong Induction

Module 10: Structural Induction, Recursive Algorithms, Induction and Recursion, Permutation and Subsets,

Module 11: Rule of Sum and Rule of Product, Bijection Rule, Permutations with Repetition, Counting Multiset, Distribute Identical Items to Distinct Objects, Distribute Identical Items to Distinct Objects with Restrictions, Principle of Inclusion and Exclusion, Combinatorial Calculations

Module 12: Lexicographic Order, Binomial Theorem, Pigeonhole Principle, Advanced Counting,

Module 13: Edges and Total Degrees, Graph Isomorphism, Graph Connectivity, Applications of Directed & Undirected Graphs, Graphs,

Module 14: Introduction to Trees, Tree Traversal, Spanning Tree, Minimum Spanning Tree,

Each module has a discussion, zyBook and a Quiz assignment.

General Description of the Subject Matter of Each Lecture or Discussion

Module 1: Logic
This module introduces the fundamentals of logic, focusing on propositions and conditional statements. Students will learn to use truth tables to evaluate compound propositions and establish logical equivalence using truth tables and propositional logic laws.

Module 2: Logic
Students will explore logical variables, predicates, and quantifiers. The module covers universal and existential quantifiers, DeMorgan's laws for negating predicates, and constructing and negating propositions with nested quantifiers.

Module 3: Logic

This module focuses on forming arguments and validating them using truth tables and inference rules. Students will learn inference rules for both propositions and quantifiers to determine argument validity.

Module 4: Proof
Students will learn direct and indirect proof techniques to verify logical statements. Emphasis is placed on constructing clear and logically sound proofs.

Module 5: Sets
This module covers basic set theory concepts, including set definitions, representations, and operations, highlighting their role in computer programs for data storage.

Module 6: Functions
Students will study functions as mappings between sets, exploring function properties, basic types, and inverse functions, with examples beyond numerical mappings.

Module 7: Relations
This module introduces relations as mappings between sets, exploring properties like reflexivity, symmetry, and transitivity. Students will also learn to represent relations using digraphs and matrices.

Module 8: Integer Properties
Students will examine integer properties, number systems (binary, octal, decimal, hexadecimal), modular arithmetic, prime factorization, GCD, LCM, and basic encryption algorithms.

Module 9: Induction and Recursion
This module covers recurrence relations and mathematical induction for validating formulas, focusing on sequences and recursive definitions.

Module 10: Induction and Recursion
Building on the previous module, students will explore recursive algorithms and extend recursive definitions to various constructs.

Module 11: Introduction to Counting
Students will learn foundational counting techniques for solving complex counting problems, including real-world applications.

Module 12: Advanced Counting
This module expands on counting techniques with algorithms for permutations and combinations, the Binomial Theorem, Pascal's Triangle, and the Pigeonhole Principle.

Module 13: Graphs
Students will explore graph theory for modeling relationships, learning graph representations and methods to identify identical graphs.

Module 14: Trees
This module introduces tree structures, traversal methods, and algorithms like BFS, DFS, and Prim's algorithm for constructing spanning and minimum spanning trees.

Required Course Syllabus Statements

Generative AI
ChatGPT (and similar Tools) in this Course:
Use ChatGPT as a learning assistant, not as a crutch. If you use it, cite it at the top of your code. You are responsible to make sure that any code or content does what it is supposed to do and says what you want it to say. Don't accept anything it generates at face value without checking it critically. These days potential employers will expect you to know how to use tools like ChatGPT to generate code, so it is a skill we need to teach you. If it helps you learn some things faster, GREAT because we can spend class time on more interesting topics. Just remember: If you REALLY want to be good, work for it.

Suggestions for using it responsibly:

If you don't have a clue, use AI to get a clue.
If you don't understand a concept, ask AI for an explanation with examples.
If some code isn't working, ask AI for help on that snippet, including broken APIs.
If you want help on improving your code, ask AI how you might improve some function or section of your code.
Tell AI to guide you toward a solution rather than giving you a solution immediately.
If you use AI, remember to note you've used it in your module docstrings and in your project submission document.
If you feel like you need smaller exercises or practice with some concept before working on some part of your project, use AI to generate exercises for you.
Whatever the AI generates, don't turn in code you could not, would not, or should not have written. That will be penalized heavily. The instructor is the judge of that.

Using Remote Testing Software
☒ This course does not use remote testing software.

☐ This course uses remote testing software. Remote test-takers may choose their remote testing locations. Please note, however, that the testing software used for this may conduct a brief scan of remote test-takers' immediate surroundings, may require use of a webcam while taking an exam, may require the microphone be on while taking an exam, or may require other practices to confirm academic honesty. Test-takers therefore shall have no expectation of privacy in their test-taking location during, or immediately preceding, remote testing. If a student strongly objects to using test-taking software, the student should contact the instructor at the beginning of the semester to determine whether alternative testing arrangements are feasible. Alternatives are not guaranteed.

Required University Syllabus Statements

Accommodations/Students with Disabilities
Students needing accommodations due to a permanent or temporary disability, pregnancy or pregnancy-related conditions may contact UVU Accessibility Services at accessibilityservices@uvu.edu or 801-863-8747.

Accessibility Services is located on the Orem Campus in BA 110.

Deaf/Hard of Hearing students requesting ASL interpreters or transcribers can contact Accessibility Services to set up accommodations. Deaf/Hard of Hearing services can be contacted at DHHservices@uvu.edu

DHH is located on the Orem Campus in BA 112.

Academic Integrity

At Utah Valley University, faculty and students operate in an atmosphere of mutual trust. Maintaining an atmosphere of academic integrity allows for free exchange of ideas and enables all members of the community to achieve their highest potential. Our goal is to foster an intellectual atmosphere that produces scholars of integrity and imaginative thought. In all academic work, the ideas and contributions of others must be appropriately acknowledged and UVU students are expected to produce their own original academic work.

Faculty and students share the responsibility of ensuring the honesty and fairness of the intellectual environment at UVU. Students have a responsibility to promote academic integrity at the university by not participating in or facilitating others' participation in any act of academic dishonesty. As members of the academic community, students must become familiar with their rights and responsibilities. In each course, they are responsible for knowing the requirements and restrictions regarding research and writing, assessments, collaborative work, the use of study aids, the appropriateness of assistance, and other issues. Likewise, instructors are responsible to clearly state expectations and model best practices.

Further information on what constitutes academic dishonesty is detailed in UVU Policy 541: *Student Code of Conduct*.

Equity and Title IX

Utah Valley University does not discriminate on the basis of race, color, religion, national origin, sex, sexual orientation, gender identity, gender expression, age (40 and over), disability, veteran status, pregnancy, childbirth, or pregnancy-related conditions, citizenship, genetic information, or other basis protected by applicable law, including Title IX and 34 C.F.R. Part 106, in employment, treatment, admission, access to educational programs and activities, or other University benefits or services. Inquiries about nondiscrimination at UVU may be directed to the U.S. Department of Education's Office for Civil Rights or UVU's Title IX Coordinator at 801-863-7999 – TitleIX@uvu.edu – 800 W University Pkwy, Orem, 84058, Suite BA 203.

Religious Accommodation

UVU values and acknowledges the array of worldviews, faiths, and religions represented in our student body, and as such provides supportive accommodations for students. Religious belief or conscience broadly includes religious, non-religious, theistic, or non-theistic moral or ethical beliefs as well as participation in religious holidays, observances, or activities. Accommodations may include scheduling or due-date modifications or make-up assignments for missed class work.
To seek a religious accommodation, a student must provide written notice to the instructor and the Director of Accessibility Services at accessibilityservices@uvu.edu. If the accommodation relates to a scheduling conflict, the notice should include the date, time, and brief description of the difficulty posed by the conflict. Such requests should be made as soon as the student is aware of the prospective scheduling conflict.

While religious expression is welcome throughout campus, UVU also has a specially dedicated space for meditation, prayer, reflection, or other forms of religious expression.