CMSI 2820: Discrete Mathematics for Computer Science

Loyola Marymount University

Professor Julian Gonzalez

Fall 2025

Course Information

Prerequisites: CMSI 1010 or ENGR 160 or ENGR 1200

Section 1: Pereira 207	**Section 2:** Pereira 207
Time: 9:55 AM – 11:35 AM	Time: 6:00 PM – 7:40 PM
Course Site: www.jag.prof	
Office: Foley Annex 139	**Email:** Julian.Gonzalez.lmu.edu

My office address, office number, and email are listed in the footer of the course website. Email is the preferred contact method. If my office door is open, feel free to stop by.

Course Description

This 4-unit course combines discrete mathematics theory with practical application in Python. Topics include Intuitionistic Propositional and Predicate Logic, Number Theory, Type Theory, Combinatorics, Graph Theory, Set Theory, and other concepts relevant to the Discrete Foundations of Computer Science.

This course meets in person in a synchronous, lecture-based format. Recordings of all lectures will be uploaded to the course site for asynchronous access, typically after class. You are encouraged to ask questions at any time during lectures.

Coursework consists of written and programming assignments, as well as optional creative projects and optional midterm and final assessments for additional credit. Collaboration is encouraged, but all submitted work must be your own original work. If you work with others or use outside sources, cite them appropriately so I can accurately assess each student's understanding.

This 4-unit course requires a minimum of 12 hours of work per week, in accordance with University policy. I encourage students to work consistently in manageable segments ("chiseling") to avoid last-minute stress. Homework assignments include checkpoints to help track progress and reinforce understanding. Additional details are provided in each assignment.

Required Materials

This course does not require students to purchase textbooks, subscriptions, paid web tools, or lab fees. You will need:

- A laptop/desktop with unrestricted terminal access
- Python 3 and an IDE (e.g., VSCode)
- Reliable internet access

All reading materials, including textbooks, articles, videos, and other media, will be available on the course site.

Learning Outcomes

This course aims to provide a rigorous foundation in the mathematics underlying Computer Science. To maintain focus, we will concentrate on core concepts essential to the field.

Many of the common problems you will solve and need to represent both conceptually and in code have distinct patterns. These distinct patterns come in many forms, but this course seeks to give you an array of knowledge and tools to work with those specifically discussed in Intuitionistic Propositional and Predicate Logic, Number Theory, Type Theory, Combinatorics, Graph Theory, and Set Theory.

While the course may not cover all topics in depth, you will develop the following core competencies:

1. Understand the fundamentals of Intuitionistic Logic, including differences from Classical Logic, and its applications in propositional and predicative reasoning, higher-order extensions, and its relationship to Type Theory.
2. Explore Boolean numbers and their role in computer logic, Venn diagrams, natural numbers, algebraic properties of operations (commutativity, associativity), and arithmetic operations such as modular arithmetic and integer division.

3. Examine fundamental collection types, including sets and tuples, their operations (union, intersection, powerset, Cartesian product), and the representation of relationships through set membership.

4. Study functions in the context of the Lambda Calculus and typed lambda calculus, including their reductions (Alpha, Beta, Gamma) and comparison to set-theoretic functions.

5. Investigate combinatorics, including combinations, permutations, factorial algebra, tetration, the pigeonhole principle, and inductive proof techniques.

6. Develop a strong understanding of graph theory, covering simple and directed graphs, trees, paths, cycles, connectivity, and graph isomorphisms, and apply these concepts to solve complex problems such as the traveling salesman problem and minimum spanning trees.

7. Recognize how previous concepts are reinterpreted through Set Theory versus Type Theory.

8. Apply Python to analyze, solve, and engage with discrete mathematics problems and algorithmic techniques.

Grading & Standards-Based System

This course uses a **standards-based grading** system. Grades are earned additively rather than averaged. You can recover points later if needed, ensuring early struggles do not lock in a lower grade.

Course Structure

The course is divided into six standards (modules), each worth 80 points (total: 480). Assignments are organized as follows:

Standard	**Assignments**	**Points**
Syllabus	HW0 (3 pts), EX S0 (2 pts)	5 (bonus)
Logic	HW1 (100 pts)	80 needed
Numbers	HW2 (100 pts), OHW1 (20 pts)	80 needed
Collections	HW3 (100 pts), OHW2 (20 pts)	80 needed
Functions	HW4 (100 pts), OHW3 (20 pts)	80 needed
Combinatorics	HW5 (100 pts), OHW4 (20 pts)	80 needed
Graph Theory	HW6 (100 pts), OHW5 (20 pts)	80 needed

Optional exams (Midterm, Final) provide additional opportunities to recover points in earlier standards.

Assignments & Policies

- **Homework:** Each standard has a primary homework (HW) and an optional homework (OHW) targeting the previous standard. Homeworks (HW) are broken down into two major areas: written and programming. The written portion will consist of theoretical problems that are best solved on paper. The programming portion will be distributed as a Python skeleton that must be completed to pass a series of unit tests that require applying our theory into practice. All homework, optional or otherwise, is due on Fridays.
- **Unexcused Extensions:** Automatic, unexcused extensions are available: turning in one portion of the Homework (written or programming) on Saturday results in a 5pt deduction. Turning in a portion on Sunday gives you the previous 5-point penalty and an additional 10-point deduction. Submitting a portion after Sunday without an excuse results in the loss of all points.
- **Excused Extensions:** Medical or personal emergencies can receive additional extensions with email communication requesting it.

Exams

The Midterm and Final are optional. They provide focused opportunities to boost scores in prior standards. If you have reached the point total required to pass a standard (80pts), then they are optional.

Submission

Assignments and exams are submitted through **Brightspace**. Grades and private materials will be posted there as well.

Course Calendar

Office Hours: Regularly held on Monday and Wednesdays, 2–5 PM in Foley Annex 139.

The following calendar lists all lectures, homework due dates, exams, and holidays. Dates are subject to change; updates will be communicated via email and Brightspace.

Date	Type	Details
Aug 26 (Tue)	Lecture	(LN0) Syllabus Day
Aug 28 (Thu)	Lecture	(LN1) Introduction to Intuitionistic logic
Sep 1 (Mon)	Holiday	Labor Day – No Class
Sep 2 (Tue)	Lecture	(LN2) Propositional Logic and Logical Connectives
Sep 4 (Thu)	Lecture	(LN3) Logical Connectives and Natural Deduction
Sep 5 (Fri)	Homework	HW0 Due
Sep 9 (Tue)	Lecture	(LN4) Introduction to Type Theory
Sep 11 (Thu)	Lecture	(LN5) Boolean Algebra and Logic
Sep 16 (Tue)	Lecture	(LN6) Abstract Algebra and Integers
Sep 18 (Thu)	Lecture	(LN7) Number Theory and Modular Arithmetic
Sep 19 (Fri)	Homework	HW1 Due
Sep 23 (Tue)	Lecture	(LN8) Tuples
Sep 25 (Thu)	Lecture	(LN9) Sets
Sep 30 (Tue)	Lecture	(LN10) Sets Continued
Oct 2 (Thu)	Lecture	(LN11) First-Order Logic and Relations
Oct 3 (Fri)	Homework	HW2 Due
Oct 7 (Tue)	Lecture	(LN12) Introduction to The Lambda Calculus
Oct 9 (Thu)	Lecture	(LN13) Reductions and Bound vs Free
Oct 10 (Fri)	Holiday	Autumn Day! No Class
Oct 12 (Sun)	Exam	Midterm Release Day
Oct 14 (Tue)	Lecture	(LN14) Formal Reductions and Function Types
Oct 16 (Thu)	Lecture	(LN15) Domain, Co-Domain, and Function Properties
Oct 17 (Fri)	Homework	HW3 Due
Oct 18 (Sat)	Exam	Midterm Due
Oct 21 (Tue)	Lecture	(LN16) Introduction to Combinatorics and Enumeration
Oct 23 (Thu)	Lecture	(LN17) Factorials, Permutations and Combinations
Oct 28 (Tue)	Lecture	(LN18) Pascal's Triangle and Double Counting Proofs
Oct 30 (Thu)	Lecture	(LN19) Bijections and The Pigeonhole Principle
Oct 31 (Fri)	Homework	HW4 Due
Nov 4 (Tue)	Lecture	(LN20) Introduction to Graph Theory
Nov 6 (Thu)	Lecture	(LN21) Connectedness and Common Graph Operations
Nov 11 (Tue)	Lecture	(LN22) Graph Types and The Traveling Salesmen
Nov 13 (Thu)	Lecture	(LN23) Forests and Minimum Spanning Trees
Nov 14 (Fri)	Homework	HW5 Due
Nov 18 (Tue)	Lecture	(LN24) Set Theory, Classical Logic, and Axioms
Nov 20 (Thu)	Lecture	(LN25) Contradiction and Induction
Nov 25 (Tue)	No Class	Thanksgiving Break

Date	Type	Details
Nov 26–28 (Wed–Fri)	Holiday	Thanksgiving
Dec 2 (Tue)	Lecture	(LN26) Applications in AI and Pure Math
Dec 4 (Thu)	Lecture	(LN27) Applications in Games and System Architecture
Dec 5 (Fri)	Homework	HW6 Due
Dec 8 (Mon)	Exam	Final Release Day
Dec 12 (Fri)	Exam	Final Due

Student Responsibilities

This page summarizes the University rules that relate to this course and our in-person classroom environment.

Academic Honesty Policy

> "Loyola Marymount University is a community dedicated to academic excellence. Academic honesty in scholarship and creative work stands at the center of LMU's academic life, and is essential for true learning and creation of knowledge to take place. As a university in the Jesuit and Marymount traditions, this community expects its members to act in accordance with the highest standards of honesty and ethics at all times. Violations of academic honesty undermine the fundamental educational mission of the University and cannot be tolerated. Students are responsible for understanding the standards of academic honesty and determining how they apply to their academic work and behavior."

Academic Honesty Policy Website

Disability Support Services

> "The mission of the Disability Support Services (DSS) Office at LMU is to provide equal access and opportunities for students with established disabilities. We are committed to promoting and celebrating the diversity of our students, staff, and faculty and work to eliminate systemic barriers, address individual bias, and maintain a respectful and equitable working environment."

Disability Support Services Website

LMU Expectation for Classroom Behavior

> "Disruptive behavior which is persistent or significantly interferes with classroom activities may be subject to disciplinary action. A student may be referred to the Office of Student Conduct and Community Responsibility if their behavior constitutes a violation of the conduct code."

OSCCR Website

Reporting of Sexual or Interpersonal Misconduct

> "Loyola Marymount University ("LMU") recognizes the significant, unacceptable and nationwide existence of Sexual and Interpersonal Misconduct on college campuses. LMU is dedicated to the prevention of such misconduct and to providing a caring, supportive and effective response when such misconduct occurs."

Misconduct Reporting Information

Communication Expectations

> Official communication must occur through LMU-approved channels (student email, Department Teams/Slack, or letters on official LMU letterhead). Students are responsible for regularly checking their LMU email. Other official channels may be audited in cases of misconduct.

Emergency Preparedness

> "Building a more resilient LMU is a shared responsibility among all students, faculty, and staff. Emergency Management serves as an all-encompassing tool for campus emergency information, disaster readiness events at LMU, and tips on how to prepare for and respond to various emergencies."

Emergency Preparedness Website

This syllabus is tentative and subject to change. Updates will be communicated promptly via email. A more detailed and expansive version of this Syllabus is provided at www.jag.prof/cmsi-2820/syllabus