Discrete Mathematics

Topic 00 — Module Introduction

Lecture 01 — Module Overview

Dr Kieran Murphy

cc BY €

Department of Computing and Mathematics,
SETU (Waterford).
(kieran.murphy@setu.ie)

Autumn Semester, 2022

Outline

- Motivation and aim of this module.
- Administration trivia — Contact hours, Assessment structure, ...
- Resources

Outline

1. Module Introduction 2
1.1. **What** is the Aim of this Module? 3
1.2. **Why** Study Discrete Mathematics? 13
1.3. **How** will the Module be Delivered? Assessed? 14
1.4. **Who** is delivering this module? 18
1.5. **When** will the module be delivered? 19
1.6. Resources 21

What? (Aim)

Aim, as per Module Descriptor*...

This module provides a solid foundation of selected topics in discrete mathematics related to computing and information sciences. The topics are covered in an elementary manner in order to reinforce understanding of concepts and improving algebraic problem-solving skills so that the student can effectively proceed with their study of a degree programme in computing.

Translation (Informal Aims)

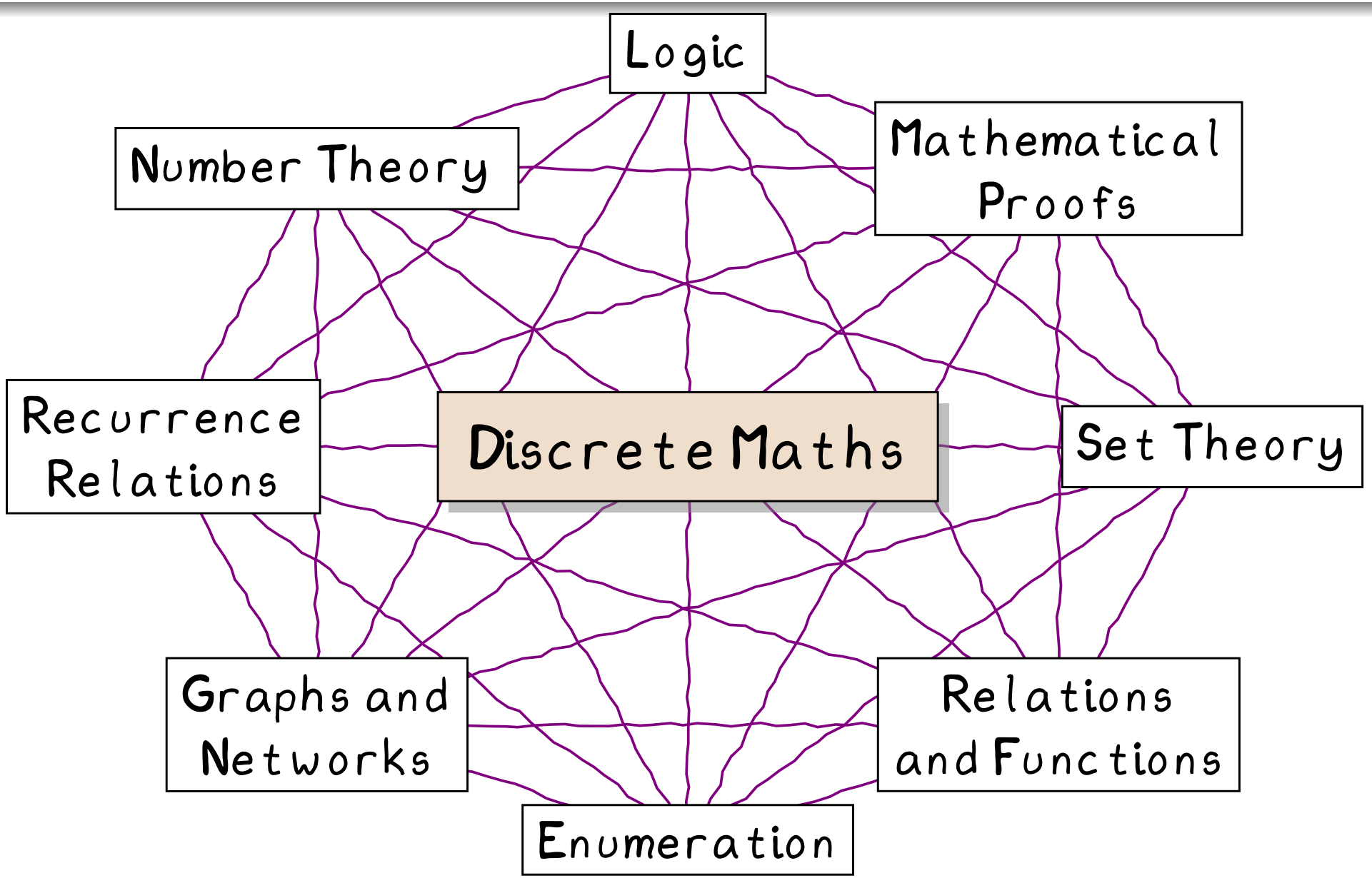
1. Reason logically — aim for precision and correctness over speed.
2. Develop and manipulate theoretical models

   — a set is a collection of things, a relation is a collection of pairs of things, a graph is a collection of things with pairwise connections, etc..
3. Translate

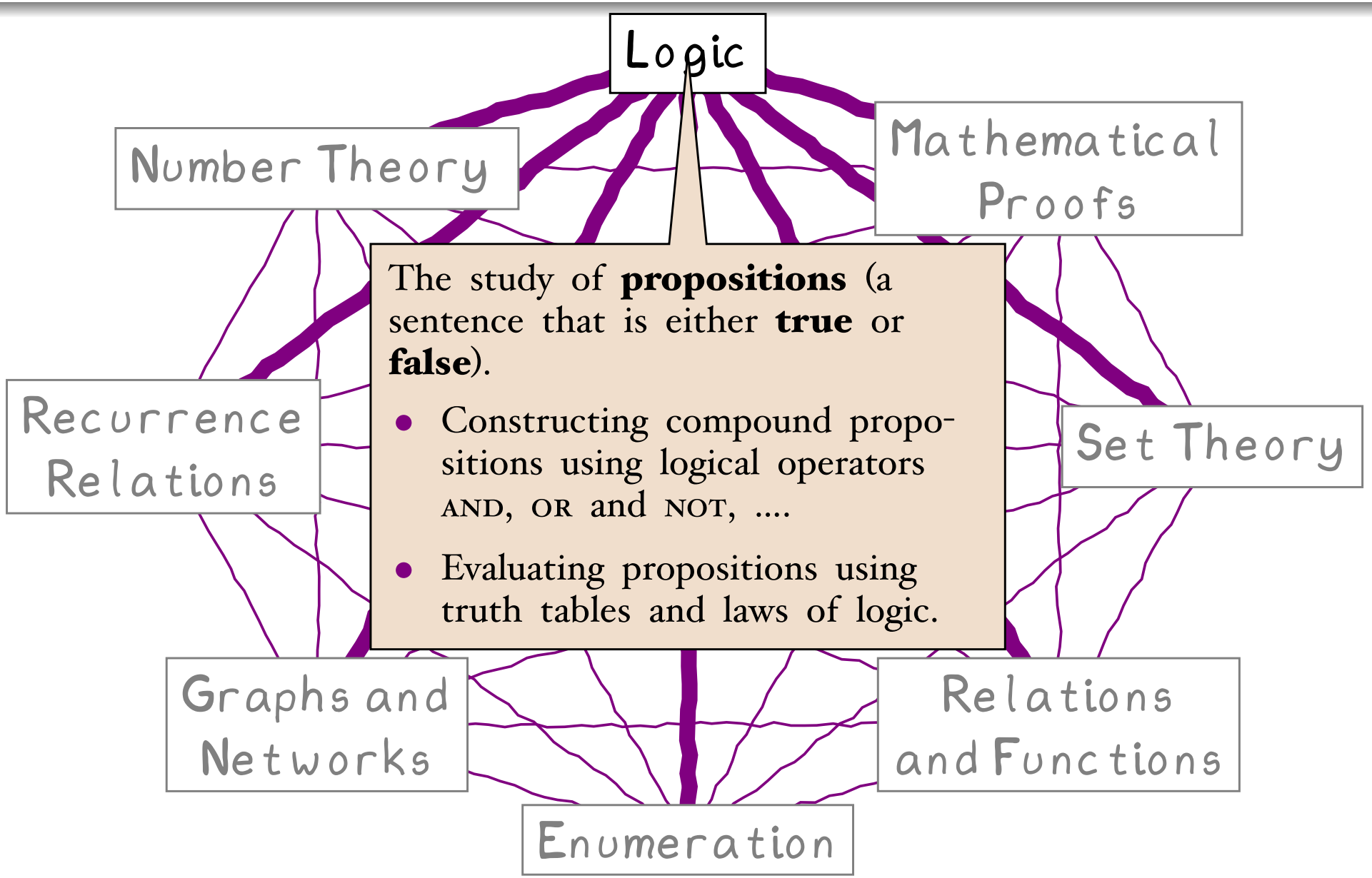
   computing concepts/languages (Python) $\leftrightarrow$ theoretical models (mathematics).

*Also, see the module descriptor for the learning outcomes, if that rocks your boat ...

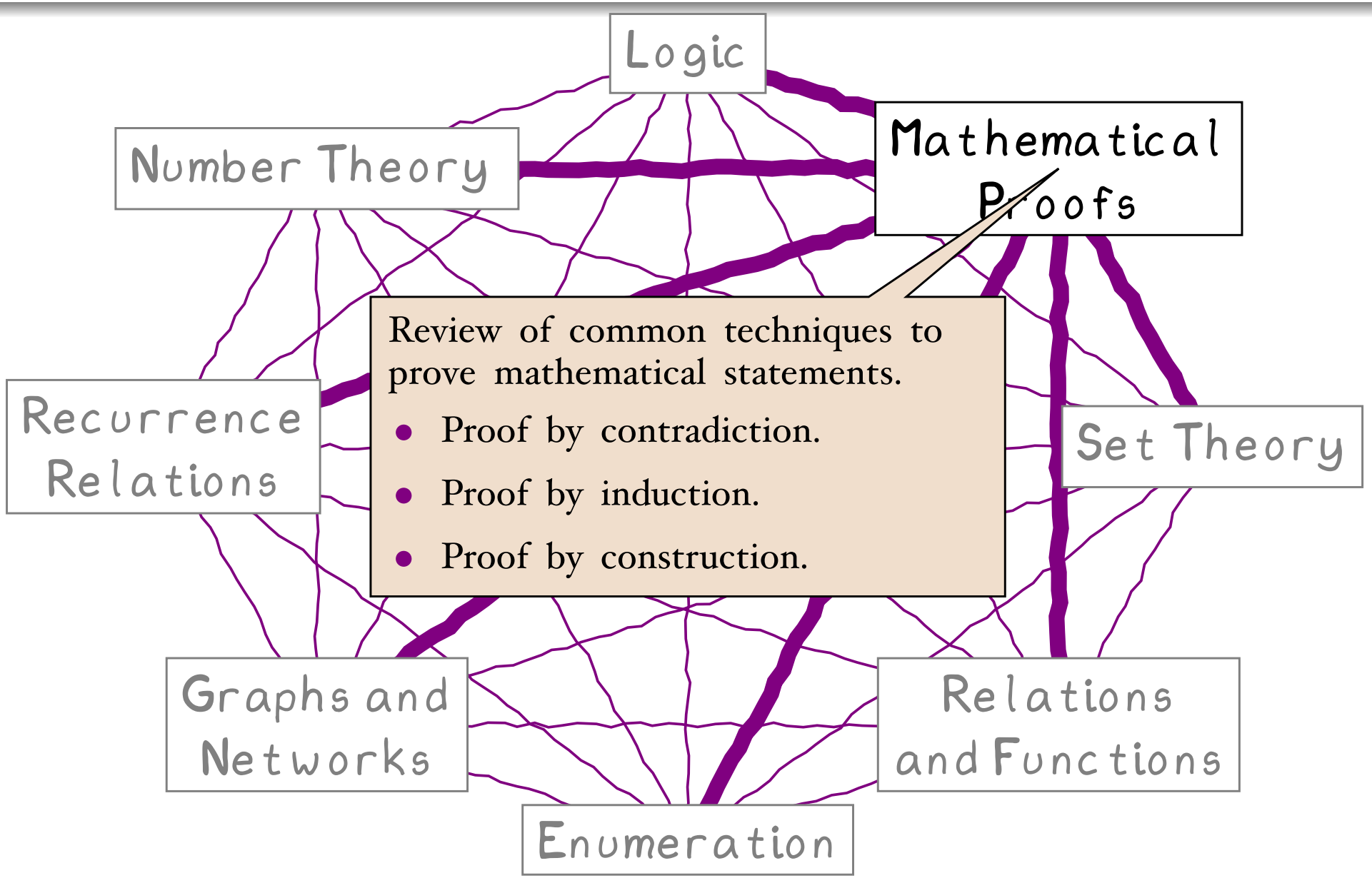
What? (Discrete Mathematics)



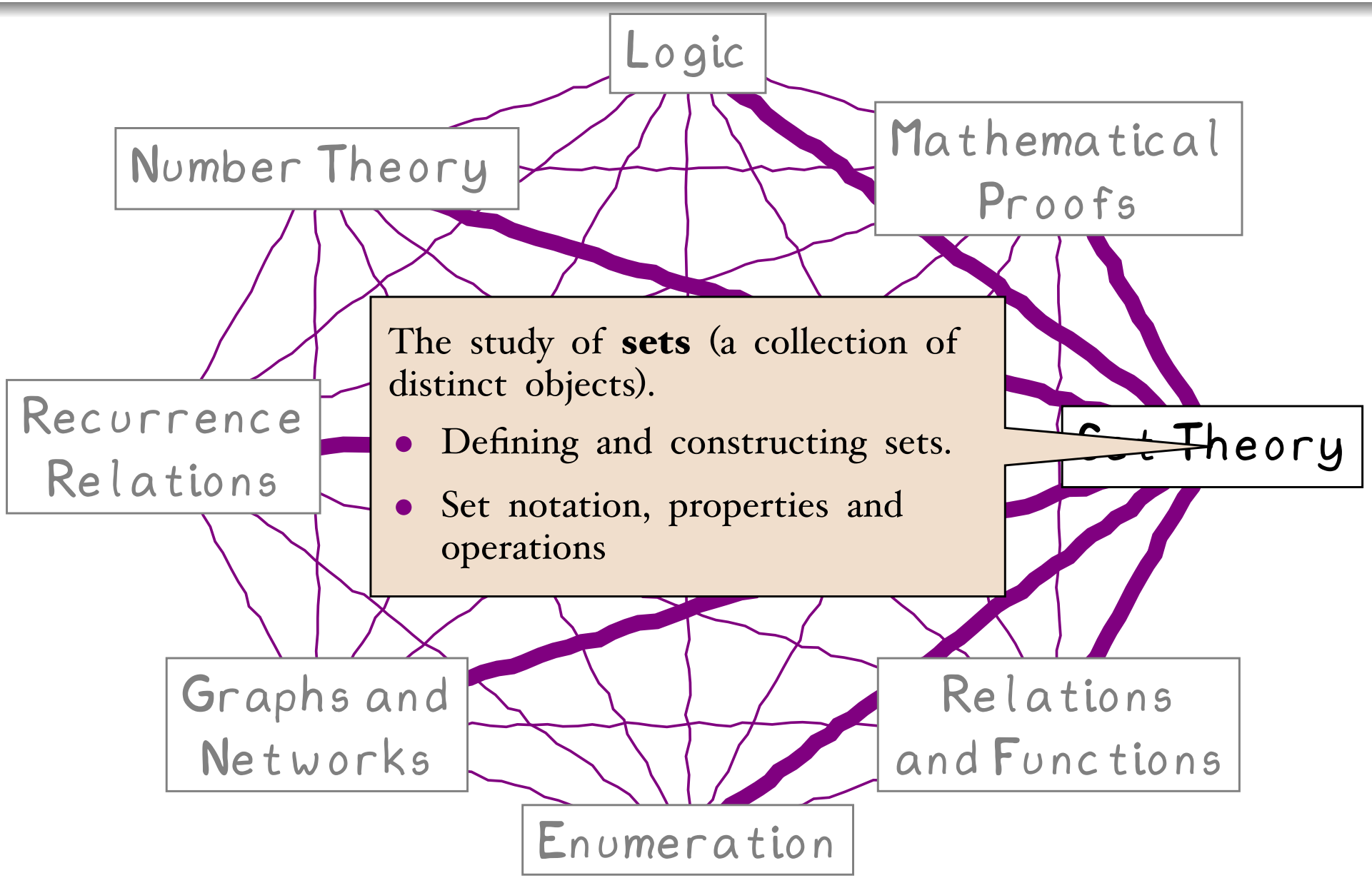
What?


Logic
Number Theory
Mathematical Proofs
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration
The study of **propositions** (a sentence that is either **true** or **false**).
• Constructing compound propositions using logical operators AND, OR and NOT, ...
• Evaluating propositions using truth tables and laws of logic.

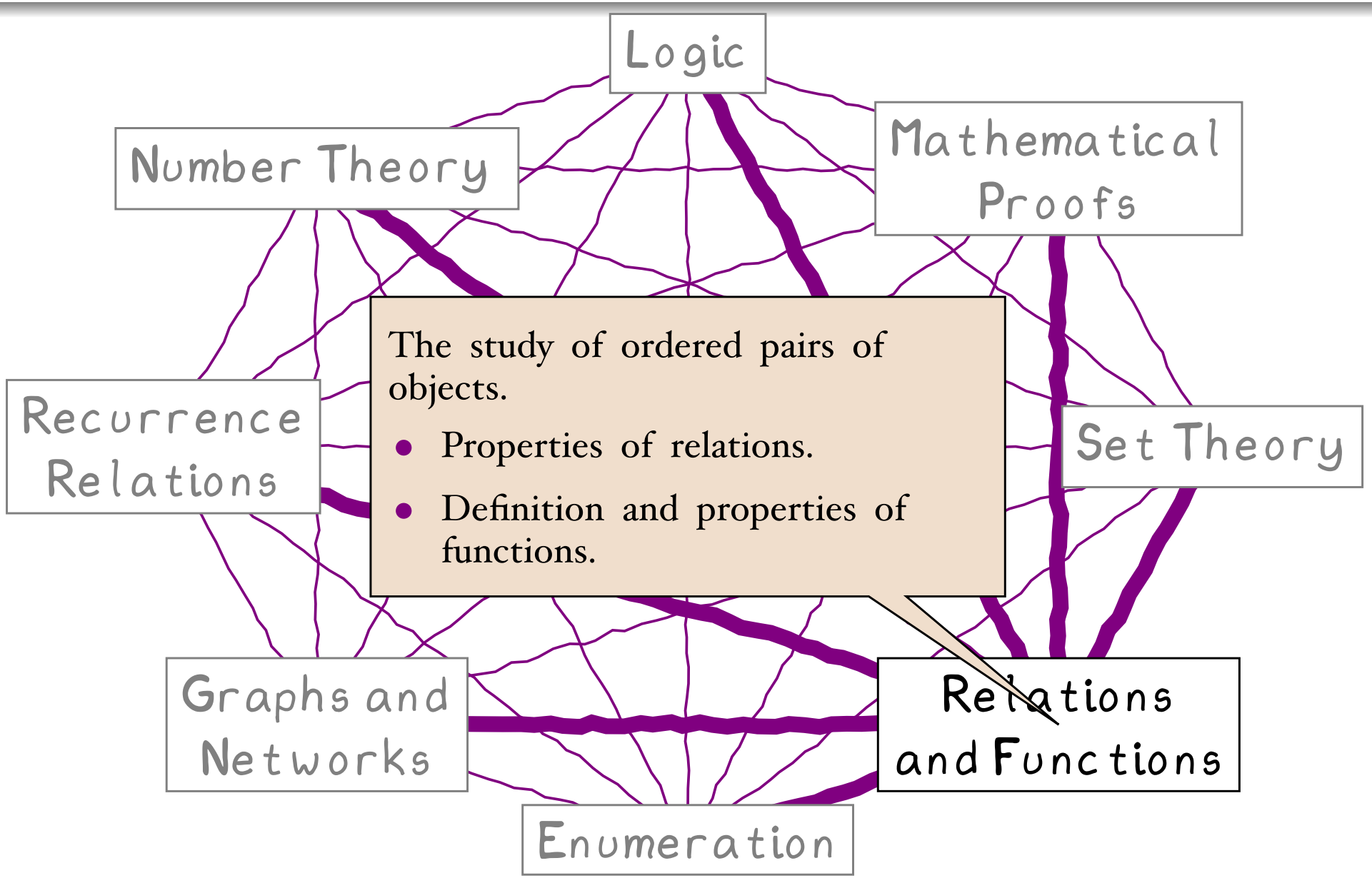
What?


Logic
Number Theory
Mathematical Proofs
Review of common techniques to prove mathematical statements.
• Proof by contradiction.
• Proof by induction.
• Proof by construction.
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

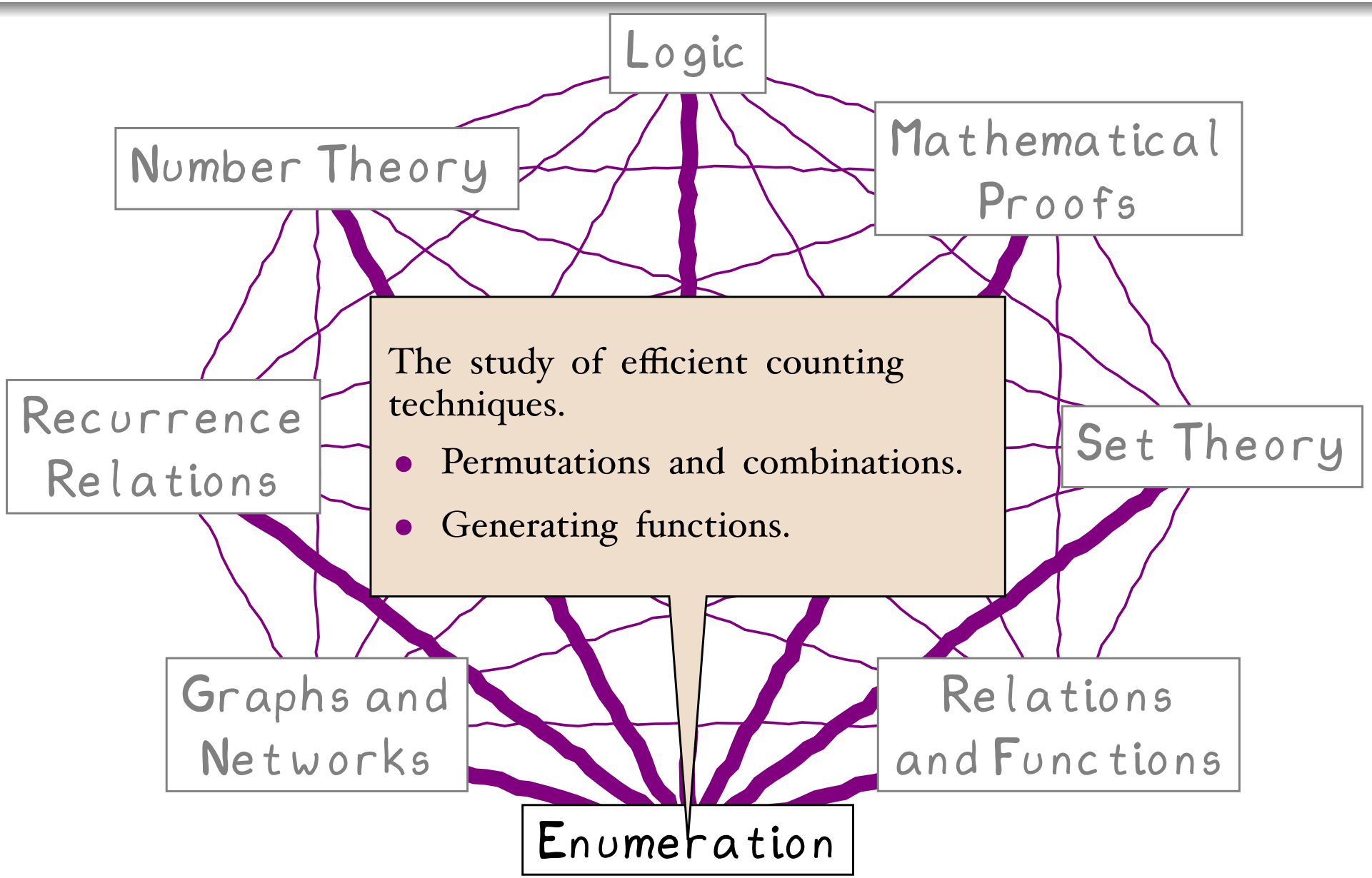
What?


Logic
Number Theory
Mathematical Proofs
The study of **sets** (a collection of distinct objects).
• Defining and constructing sets.
• Set notation, properties and operations
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

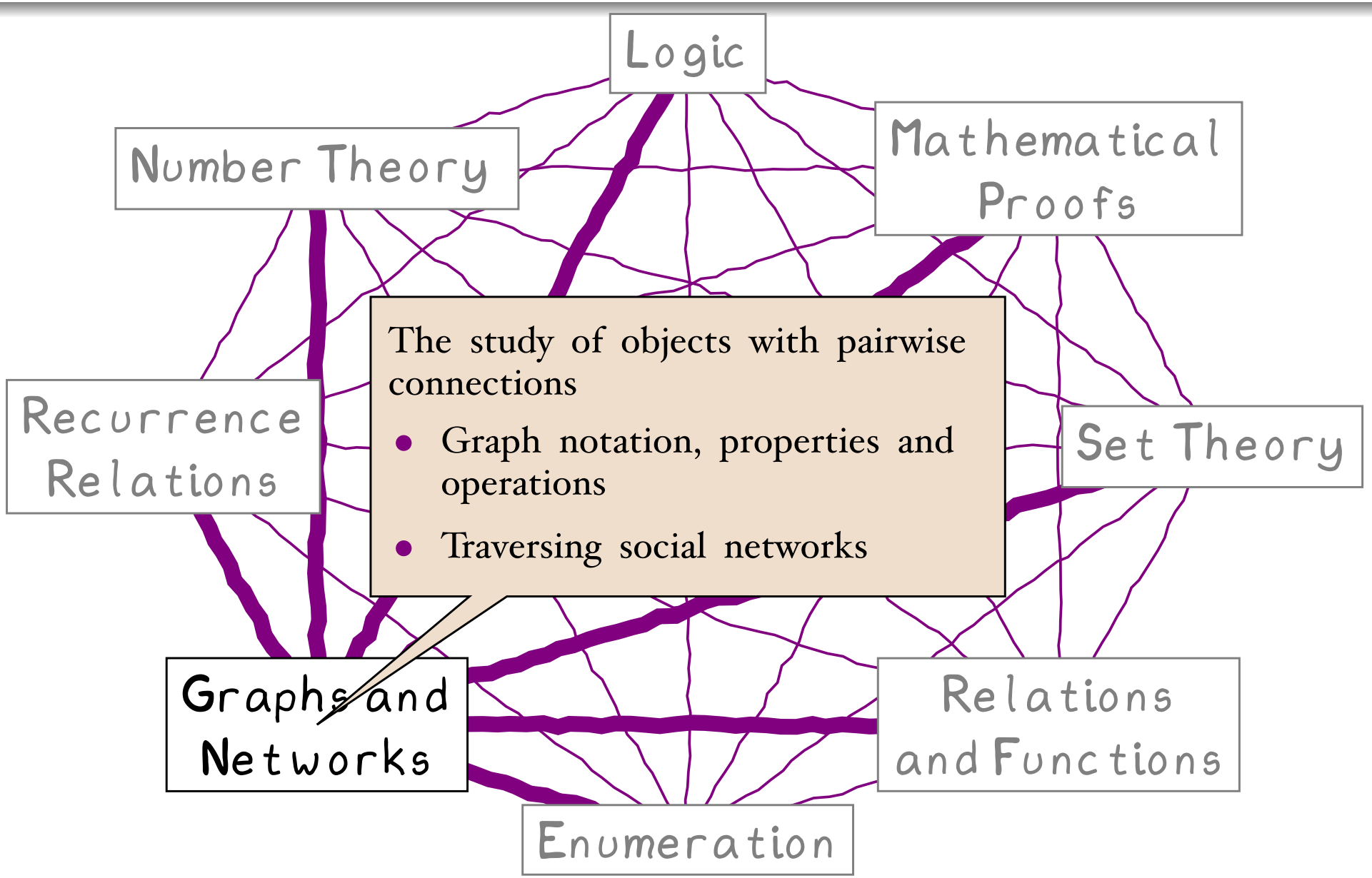
What?


Logic
Number Theory
Mathematical Proofs
The study of ordered pairs of objects.
Properties of relations.
Definition and properties of functions.
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

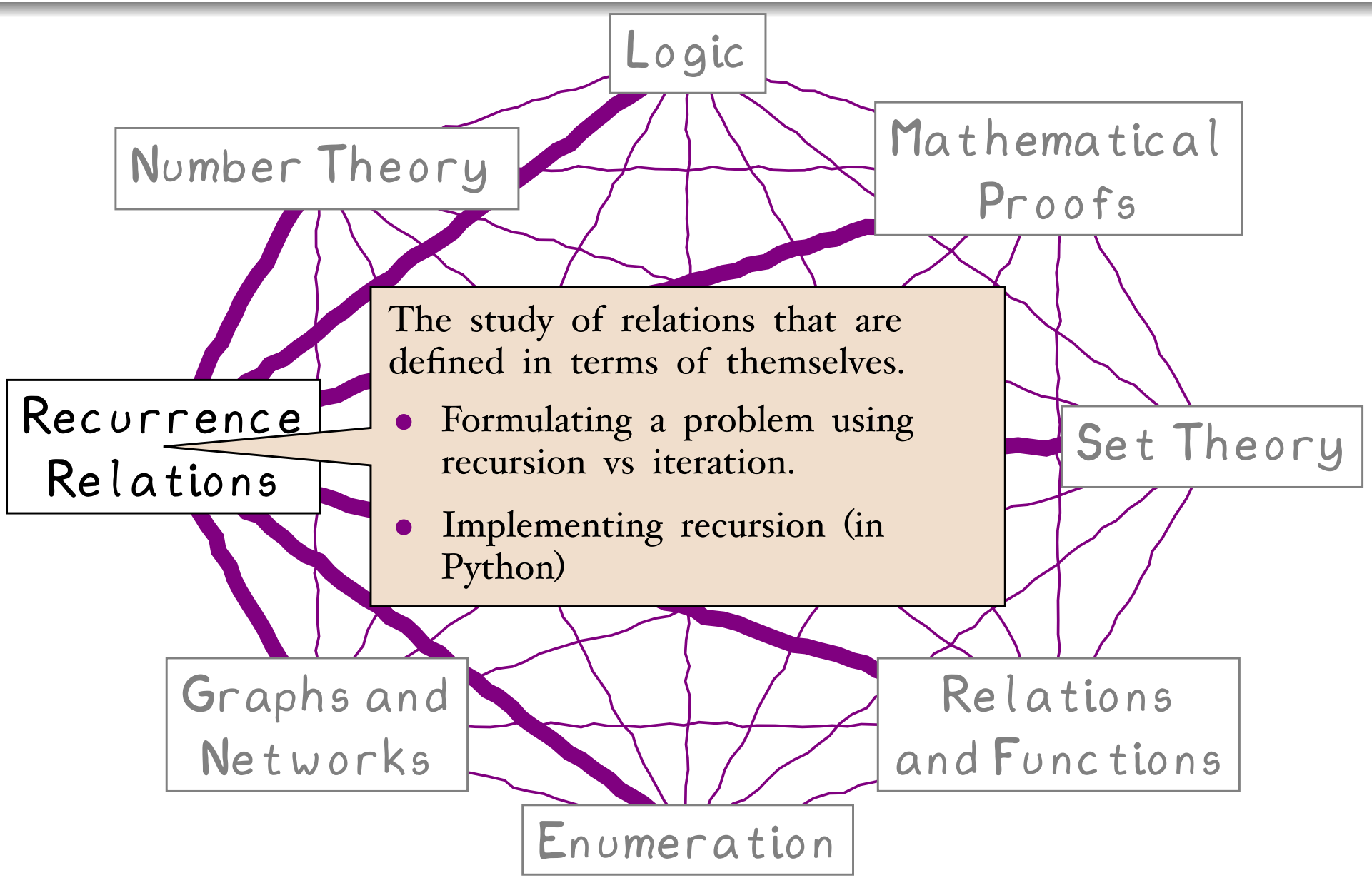
What?


Logic
Number Theory
Mathematical Proofs
The study of efficient counting techniques.
Permutations and combinations.
Generating functions.
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

What?


Logic
Number Theory
Mathematical Proofs
The study of objects with pairwise connections
Graph notation, properties and operations
Traversing social networks
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

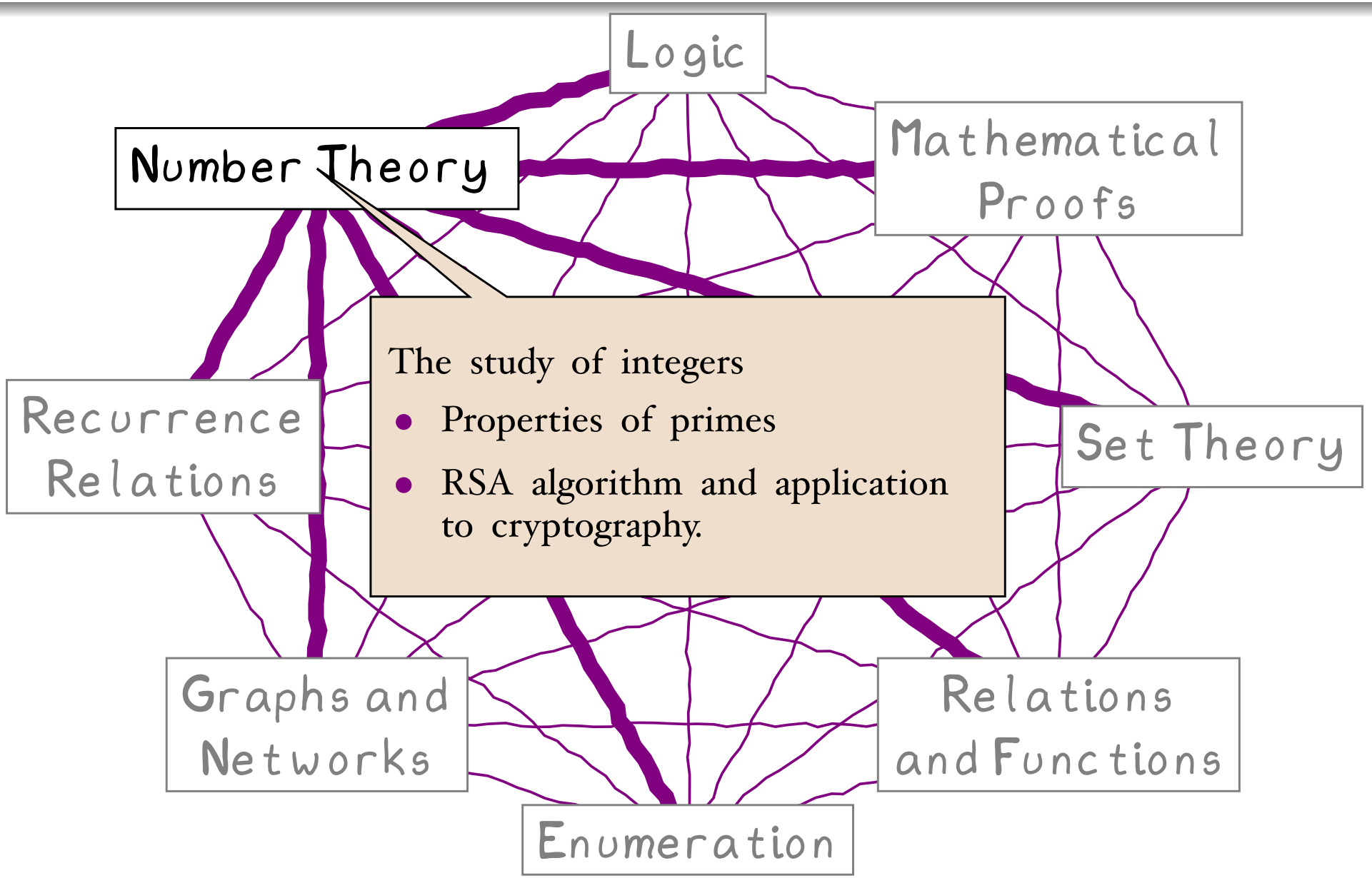
What?



The study of relations that are defined in terms of themselves.

- Formulating a problem using recursion vs iteration.
- Implementing recursion (in Python)

What?


Logic
Number Theory
Mathematical Proofs
The study of integers
Properties of primes
RSA algorithm and application to cryptography.
Recurrence Relations
Set Theory
Graphs and Networks
Relations and Functions
Enumeration

Why?

Many reasons . . . to pick one . . .

Machine learning is the future of computing

Discrete mathematics is the core of machine learning.

*The AI Revolution: Our Immortality or Extinction
waitbutwhy.com/2015/01/artificial-intelligence-revolution-2.html

Why?

Many reasons . . . to pick one . . .

Machine learning is the future of computing

Discrete mathematics is the core of machine learning.

*The AI Revolution: Our Immortality or Extinction
waitbutwhy.com/2015/01/artificial-intelligence-revolution-2.html

Why?

Many reasons . . . to pick one . . .

Machine learning is the future of computing

Discrete mathematics is the core of machine learning.

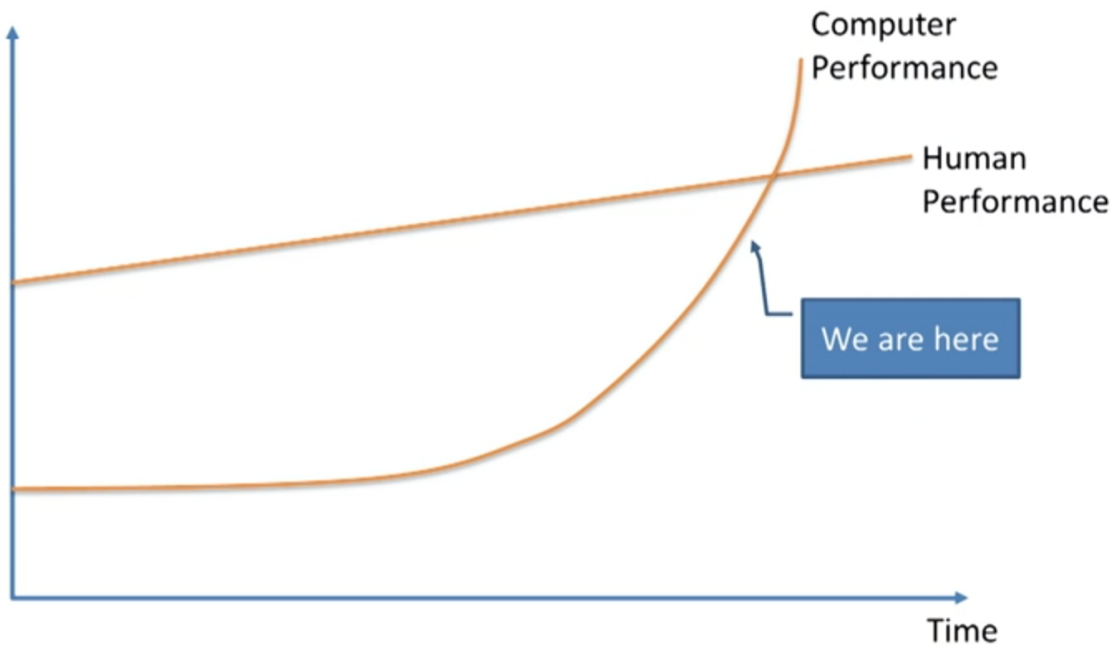
*The AI Revolution: Our Immortality or Extinction
waitbutwhy.com/2015/01/artificial-intelligence-revolution-2.html

Why?

Many reasons . . . to pick one . . .

Machine learning is the future of computing

Discrete mathematics is the core of machine learning.



*The AI Revolution: Our Immortality or Extinction
waitbutwhy.com/2015/01/artificial-intelligence-revolution-2.html

How? (Contact Hours)

- Three lectures per week
  - Cover concepts, definitions, examples, etc.
  - BUT feel free to stop me and ask questions at any point.
  - **You need to have printout of notes in advance of lecture.**
  - Ideally you skimed over the notes in advance of lecture.
  - **Take notes during lectures.**
- One tutorial per week
  - Review of exercises based on the material covered in the lecturers.
  - **You need to have printout of tutorial sheets in advance of lecture.**
  - Ideally you have attempted/completed some/all questions in advance of tutorial and you are just attending the tutorials to show off.
  - Online quiz for self review at end of each topic.
- One practical per week
  - Using Python (via *jupyter notebooks*) to demonstrate implementation details of discrete mathematics concepts.
  - Introduce programming in Python — never have too much programming.
  - **You need to upload notebook by end of week (Saturday 11:00pm).**

How? (Contact Hours)

- Three lectures per week
  - Cover concepts, definitions, examples, etc.
  - BUT feel free to stop me and ask questions at any point.
  - **You need to have printout of notes in advance of lecture.**
  - Ideally you skimed over the notes in advance of lecture.
  - **Take notes during lectures.**
- One tutorial per week
  - Review of exercises based on the material covered in the lecturers.
  - **You need to have printout of tutorial sheets in advance of lecture.**
  - Ideally you have attempted/completed some/all questions in advance of tutorial and you are just attending the tutorials to show off.
  - Online quiz for self review at end of each topic.
- One practical per week
  - Using Python (via *jupyter notebooks*) to demonstrate implementation details of discrete mathematics concepts.
  - Introduce programming in Python — never have too much programming.
  - **You need to upload notebook by end of week (Saturday 11:00pm).**

How? (Contact Hours)

- Three lectures per week
  - Cover concepts, definitions, examples, etc.
  - BUT feel free to stop me and ask questions at any point.
  - **You need to have printout of notes in advance of lecture.**
  - Ideally you skimed over the notes in advance of lecture.
  - **Take notes during lectures.**
- One tutorial per week
  - Review of exercises based on the material covered in the lecturers.
  - **You need to have printout of tutorial sheets in advance of lecture.**
  - Ideally you have attempted/completed some/all questions in advance of tutorial and you are just attending the tutorials to show off.
  - Online quiz for self review at end of each topic.
- One practical per week
  - Using Python (via *jupyter notebooks*) to demonstrate implementation details of discrete mathematics concepts.
  - Introduce programming in Python — never have too much programming.
  - **You need to upload notebook by end of week (Saturday 11:00pm).**

How? (Assessment Structure)

75% End of Semester Exam

Current plan (this is subject to change so ask about this in week 10!)

- 4 questions (typically 3–5 parts per question. Answer all questions (i.e. no choice).
- Tend not to have question per topic.
- Same material as last year — see pre-Covid exam papers, but there may still be some differences in format/style of questions as will the relative emphasis/weighting of the different topics.

25% Continuous Assessment

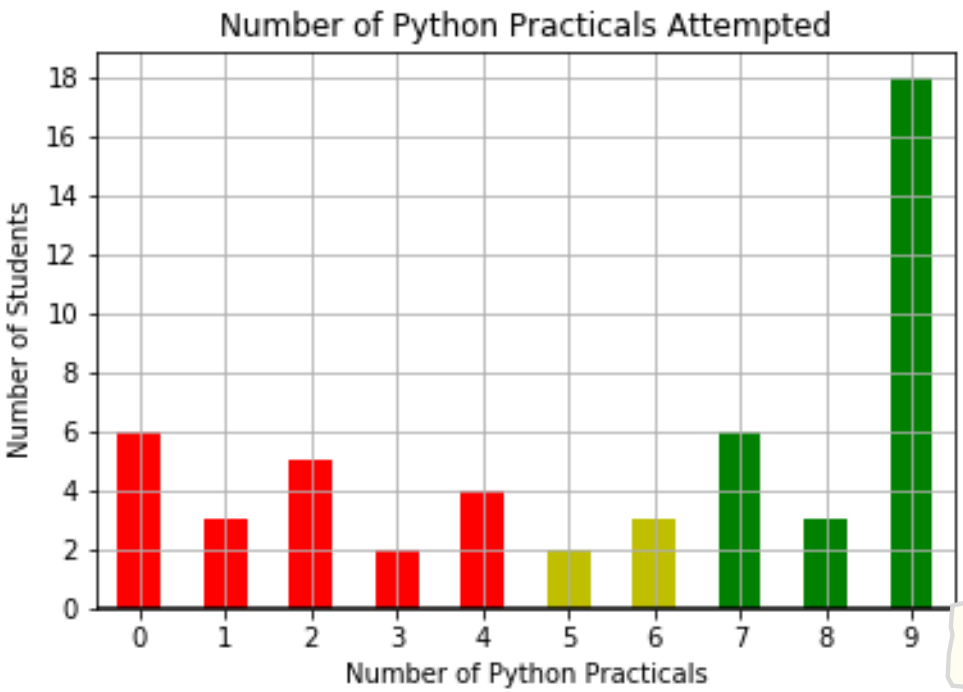
- Practical work based on 10 python practicals and 2 online class tests[†].
- In theory[‡], weekly assignments, are graded in advance of next week.

[†] The end of topic online quiz are formative assessment only (zero weighting).

[‡] In practice, I ~~may~~ will fall behind a bit.

A Brief Look at 2019/20 Results

- 52 students enrolled, but only 28 passed! $\Longrightarrow$ pass rate of 53.8%.
- Of the 32 students who attempted at least 5 practicals, 25 passed
  $\Longrightarrow$ pass rate of 78.1%.
- Of the 27 students who attempted at least 7 practicals, 24 passed
  $\Longrightarrow$ pass rate of 88.8%.



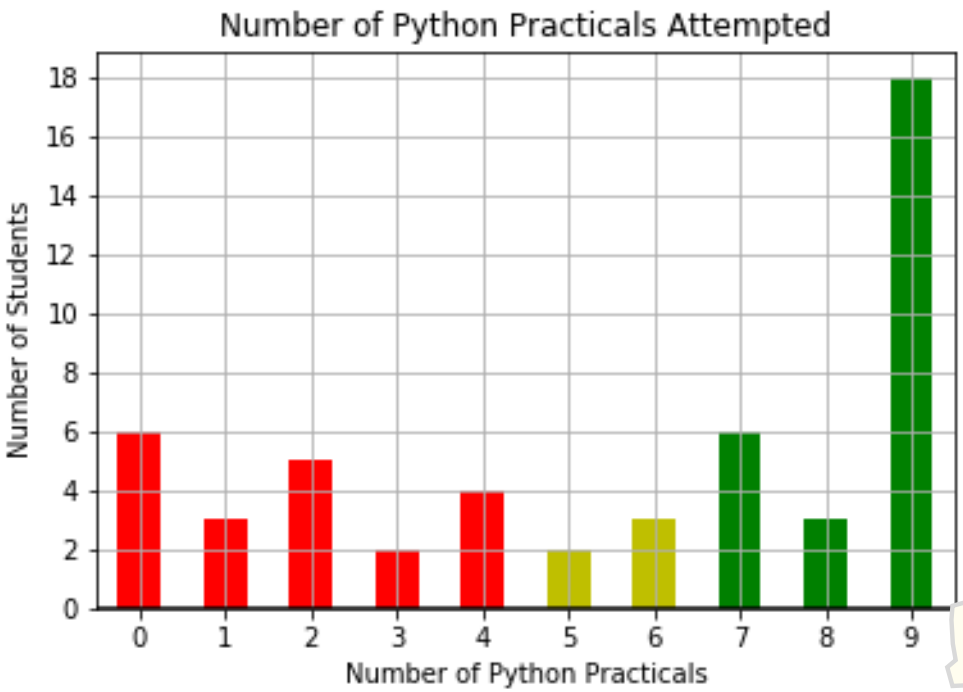
Keep up with the material:

- Read notes before & after lecturers
- Attend practicals
- Attempt tutorial questions.

Note "Attempted" $\neq$ "Completed correctly"

A Brief Look at 2019/20 Results

- 52 students enrolled, but only 28 passed! $\Longrightarrow$ pass rate of 53.8%.
- Of the 32 students who attempted at least 5 practicals, 25 passed
  $\Longrightarrow$ pass rate of 78.1%.
- Of the 27 students who attempted at least 7 practicals, 24 passed
  $\Longrightarrow$ pass rate of 88.8%.



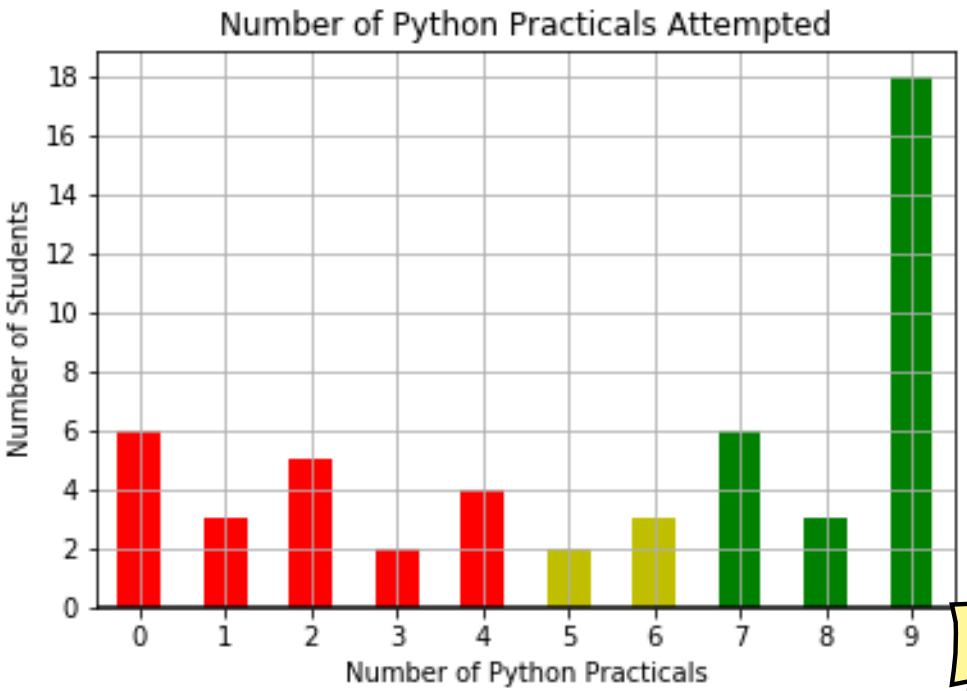
Keep up with the material:

- Read notes before & after lecturers
- Attend practicals
- Attempt tutorial questions.

Note "Attempted" $\neq$ "Completed correctly"

A Brief Look at 2019/20 Results

- 52 students enrolled, but only 28 passed! $\Longrightarrow$ pass rate of 53.8%.
- Of the 32 students who attempted at least 5 practicals, 25 passed
  $\Longrightarrow$ pass rate of 78.1%.
- Of the 27 students who attempted at least 7 practicals, 24 passed
  $\Longrightarrow$ pass rate of 88.8%.



Keep up with the material:

- Read notes before & after lecturers
- Attend practicals
- Attempt tutorial questions.

Note "Attempted" $\neq$ "Completed correctly"

A Brief Look at Last Year's Results (2022/23)

- 88 students, pass/compensation rate of 64%. (better but can still improve)
- Comparison of Practical Work (CA) vs Final Exam:
  - One student passed while failing the CA

    It is possible to pass this module without doing the practical work but the odds are against you.
  - Average grade on CA was 62.8% while on Final Exam was 47.5%

    The CA is graded easier (a carrot to help keep you motivated during the semester).
  - Average number of practicals attempted last year was 6.6.

    While (like in 2019) the more practicals a student attempted the more likely they passed the module, if you miss a week or two it is not "end of the world".

A Brief Look at Last Year's Results (2022/23)

- 88 students, pass/compensation rate of 64%. (better but can still improve)
- Comparison of Practical Work (CA) vs Final Exam:
  - One student passed while failing the CA

    It is possible to pass this module without doing the practical work but the odds are against you.
  - Average grade on CA was 62.8% while on Final Exam was 47.5%

    The CA is graded easier (a carrot to help keep you motivated during the semester).
  - Average number of practicals attempted last year was 6.6.

    While (like in 2019) the more practicals a student attempted the more likely they passed the module, if you miss a week or two it is not "end of the world".

Who?

Dr Denis Flynn

Room 327
051–30 2068
Denis.Flynn@setu.ie

Background

- PhD in Applied Mathematics and Civil Engineering.
- BSc (H) Physics.

Academic Interests

- Dynamical systems, in particular systems with hysteresis
- Game development
- Langauges: C/C++, Python, Java

Dr Kieran Murphy

Room 313
051–30 2055
Kieran.Murphy@setu.ie

Background

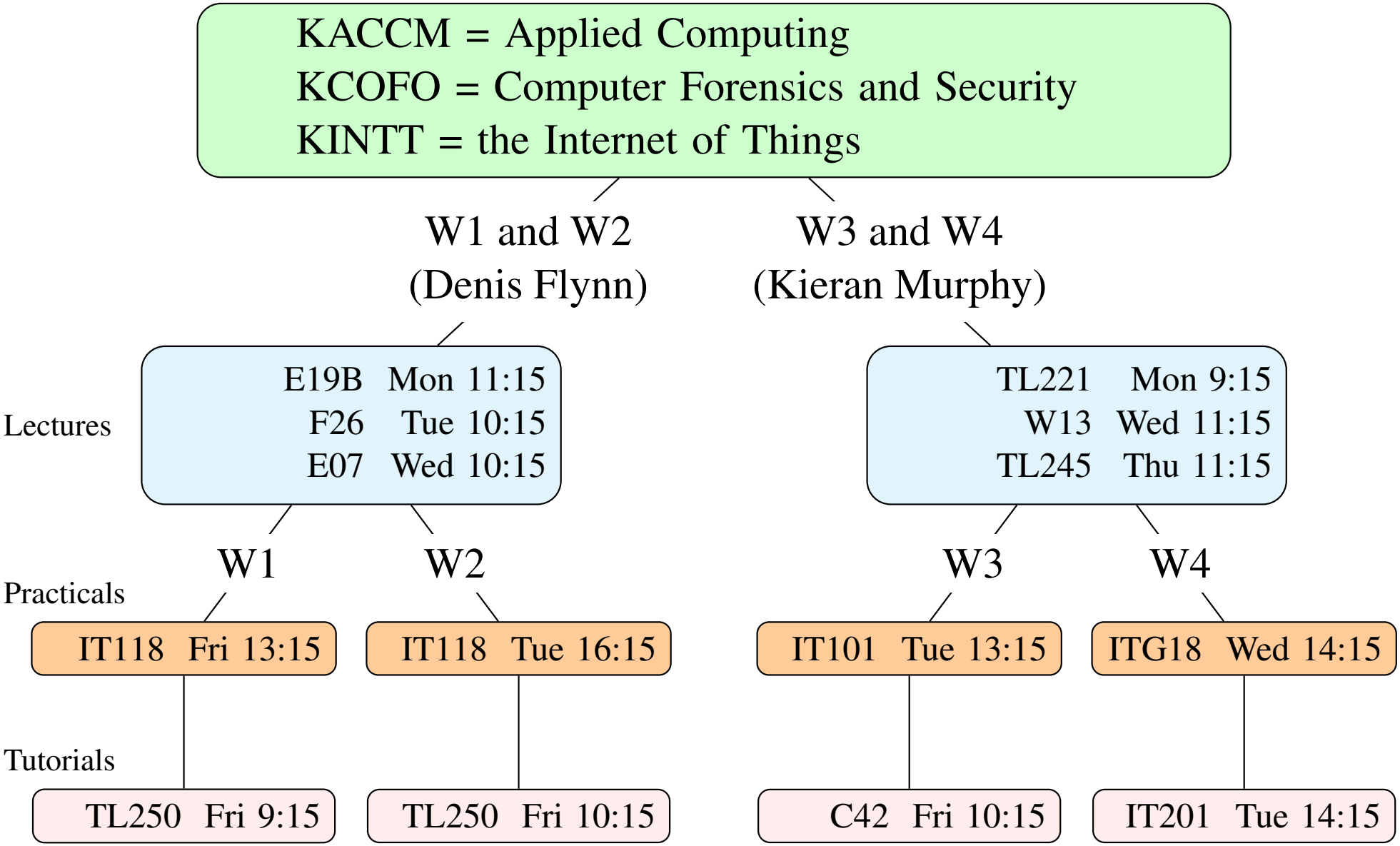
- PhD in Applied Mathematical Sciences
- BSc (H) Applied Mathematics.

Academic Interests

- Dynamical systems, in particular numerical analysis
- Game development
- Langauges: C/C++, Python, Java

When?

(Correct as of 19 Sep 2022)



Proposed Calendar[§]

Week	Topic	Practical
0	Intro half-week (1 lecture)	P00 — Introduction to Python and Colab
1	Logic	P01 — Propositional logic in Python
2	Logic	P02 — Mathematical Proofs in Python
3	Set Theory	P03 — Sets in Python
4	Set Theory	P04 — Applications of Sets
	Reading Week	
5	Relations and Functions	P05 — Implementating Relations in Python
6	Relations and Functions	P06 — Functions and their Properties
7	Enumeration	P07 — Implementing Counting Techniques
8	Enumeration	P08 — Permutations and Combinations
9	Graphs and Networks	P09 — Graph Generation using Networkx
10	Recurrence Relations	P10 — Recursion
10	Module Review (2 lectures)	P07 — Enumeration in Python
11/12	End of Semester Exams	

[‡]Week 1 starts on Monday 26 September, etc. Please note that this is provisional and will be updated during the semester. In fact, as it stands it is optimistic in assuming that we won't have a break due to Covid, Monkeypox, Ebola, . . . but let's start off being optimistic.

Resources

- URL: moodle.wit.ie/course/view.php?id=191387
- Used for all notices, assignment and practical work submissions.

- URL: SETU-DiscreteMathematics.github.io/live
- Used for all content (slides, notebooks, tutorial sheets).

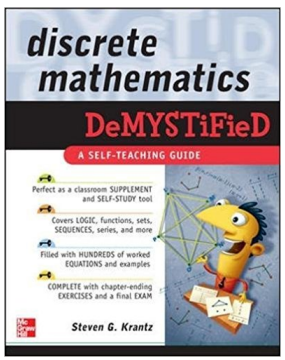
- URL: discretemathe-pb73185.slack.com
- Used for instant messaging, one-on-one sessions, etc.

Software

We will use the online Google colab environment for python, but if you want to install python on your laptop you could use the anaconda distribution from www.anaconda.com (just install the 64-bit, version 3.+).

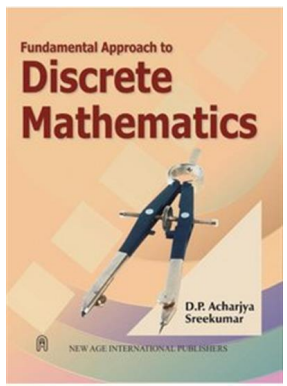
Text Books

I like the following textbooks on discrete mathematics and expect that my notes will overlap significantly with these books. I do encourage you to read[¶] them[‖], however, be aware they may use different notation or cover different topics.



Discrete Mathematics Demystified
by Steven Krantz
Touches on nearly all of the topics that we hope to cover. We will probably go into greater depth in places, but a very nice and short read.

Fundamental Approach to Discrete Mathematics
by D. P. Acharjya Sreekumar
I also liked this book, however, due to time constraints, this module only focuses on material in chapter 1–4, 8, and 10.



[¶] or skim them over a coffee or two.

[‖] I also like *Applied Discrete Structures* by Alan Doerr and Kenneth Levasseur — it is a good source of exercises. (and is free (legally))

Final Comments on Module

- Discrete Mathematics concepts appear either directly or indirectly in approximately 22 of the 30 modules on your degree.
  $\Longrightarrow$ *Knowing Discrete Mathematics concepts greatly simplfies rest of the course.*
- The module is intended to be an introduction to a large number of topics, so treatment is broad rather then deep.
  - ✔ Most of material is at an introductory level.
  - ⚠ Keeping in sync with material, practicals and tutorials is important.
- The continuous assessment (the practicals) is intended to reenforce the connections between programming and discrete mathematics.

The CA is a "carrot not a stick" — we want you to enjoy the module and keep up to date with the material.