Discrete Structures: CMPSC 102

Oliver BONHAM-CARTER

Fall 2022

Week 1

The Class Websites

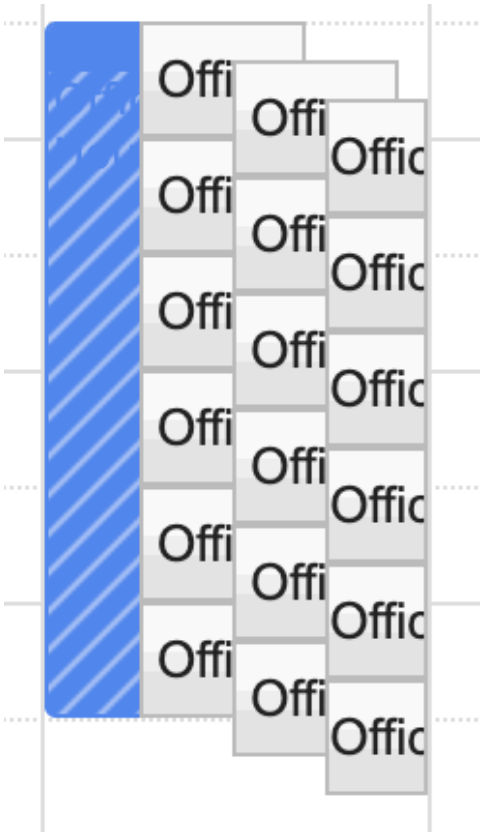
General Information

- The course Website:
  - `https://www.oliverbonhamcarter.com/classes/discretestructures/`

The Class Website

Office hours

- Booking office hours:
  - `https://www.oliverbonhamcarter.com/contactandabout/`



The Class Website

Please be familiar with the course syllabus

- Check the syllabus
  - `https://github.com/CMPSC-102-Allegheny-College-Fall-2022/classDocs/blob/main/README.md`

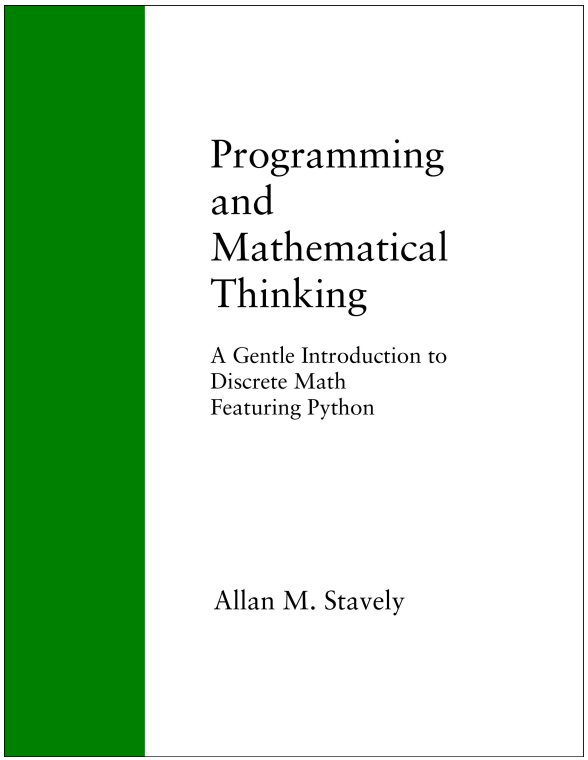
Figure: Did I search for *Syllabus* correctly?

Class and lab meeting times

Please read the syllabus before next class!!

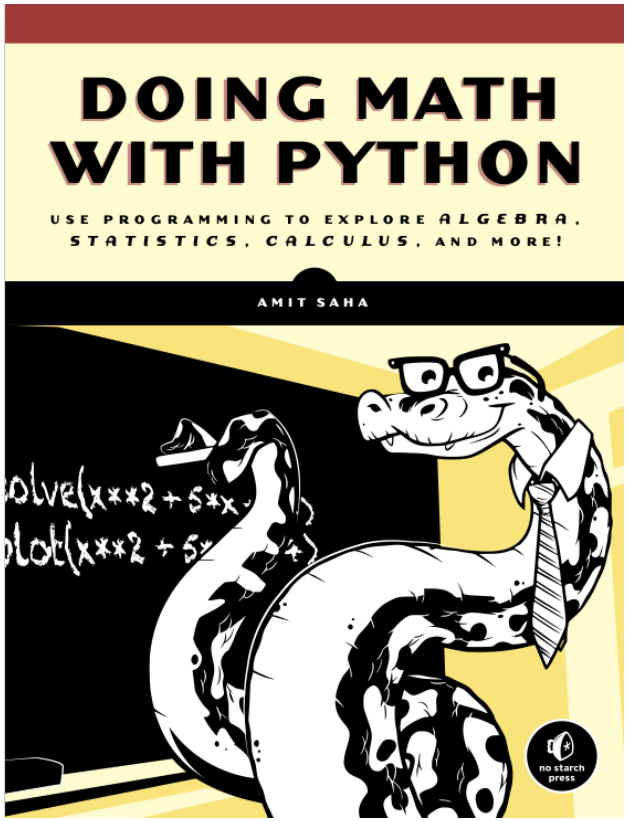
- **Lecture, Discussion, Presentations, and Group Work:**
  - Monday, Wednesday, Friday 1:30pm - 2:20pm, Alden Hall 101
- **Laboratory Session:**
  - Tuesday 2:30PM - 4:20PM, Alden Hall 101

Textbook



- *Programming and Mathematical Thinking - A Gentle Introduction to Discrete Math Featuring Python* by Allan M. Stavely; ISBN paperback 978-1-938159-00-8 and ISBN ebook: 978-1-938159-01-5

Textbook



- *Doing Math with Python* by Amit Saha; ISBN paperback: 1-59327-640-0

Learning as a Computer Scientist?

In terms of programming

Key Question

How do I connect **mathematical terminology** *(e.g., mapping, function, number, sequence, and set), to the implementation of* **Python programs** *that declare and call functions and declare and manipulate variables?*

Learning Objectives

To **remember** *and* **understand** *some of the discrete mathematics and Python programming concepts, setting the stage for the exploration of discrete structures.*

Learning as a Computer Scientist?

For example

Discrete Structures = Math + Code

- **Discrete mathematics**
  - P Made up from: *symbols*, *character strings*, *truth values*, *objects*, and *collections of these entities* as stored in *sets* or *tuples* (for example)
  -
- S Specifying and designing a **computer program**
  - Describe input, output, and internal objects
  - Use the vocabulary of discrete mathematics
  - Implement and test the program in a language
- **Our goal**:
  - To implement a program P that meets a particular specification S

Learning as an Analytical Thinker?

In terms of mathematics

> *"An introduction to the foundations of computer science with an emphasis on understanding the abstract structures used to represent discrete objects."*

Wait! What?

We keep using the word, **discrete**. What do we mean here?

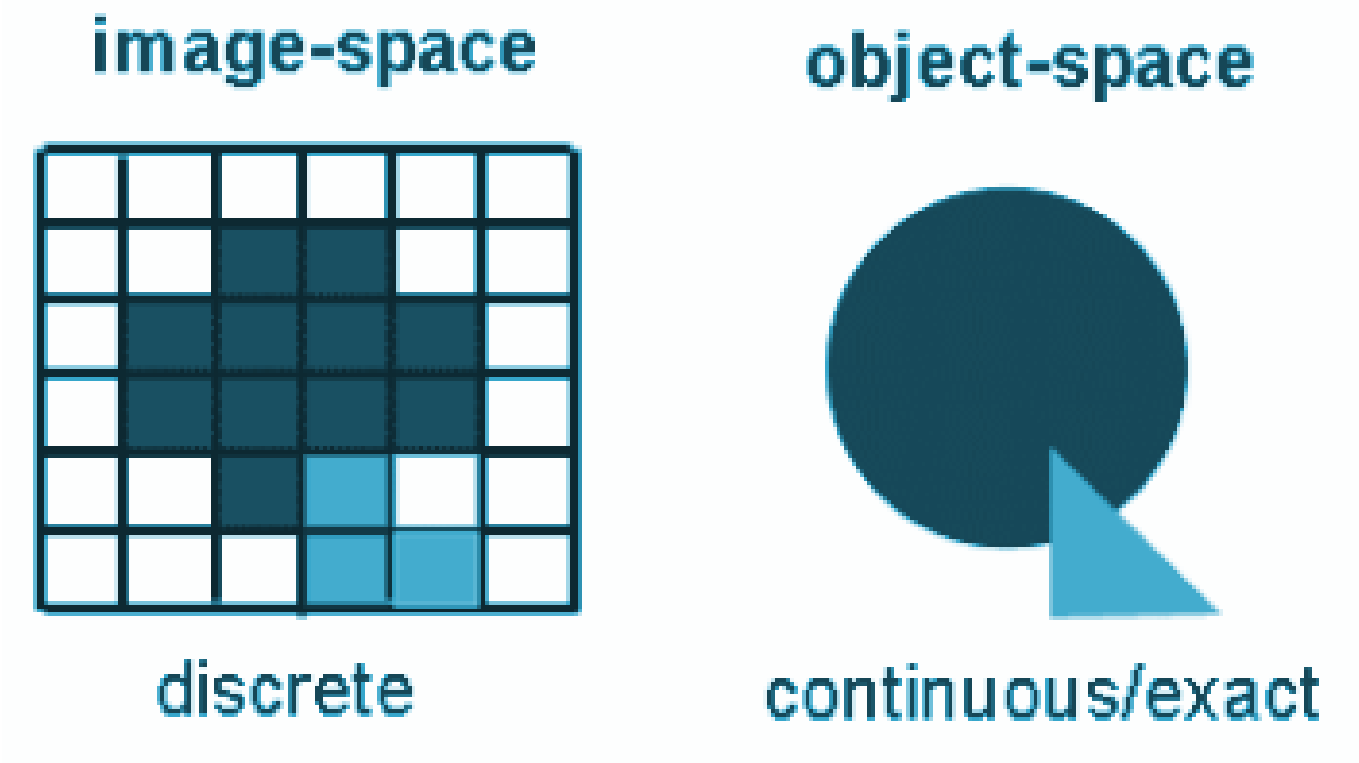
Discreet or *Discrete*

- **Discreet** means *unobtrusive* or *unnoticeable* (not this course!)
- **Discrete** means *separate*, not continuous or *not sharing any common space*

Discrete and *Countable* Objects

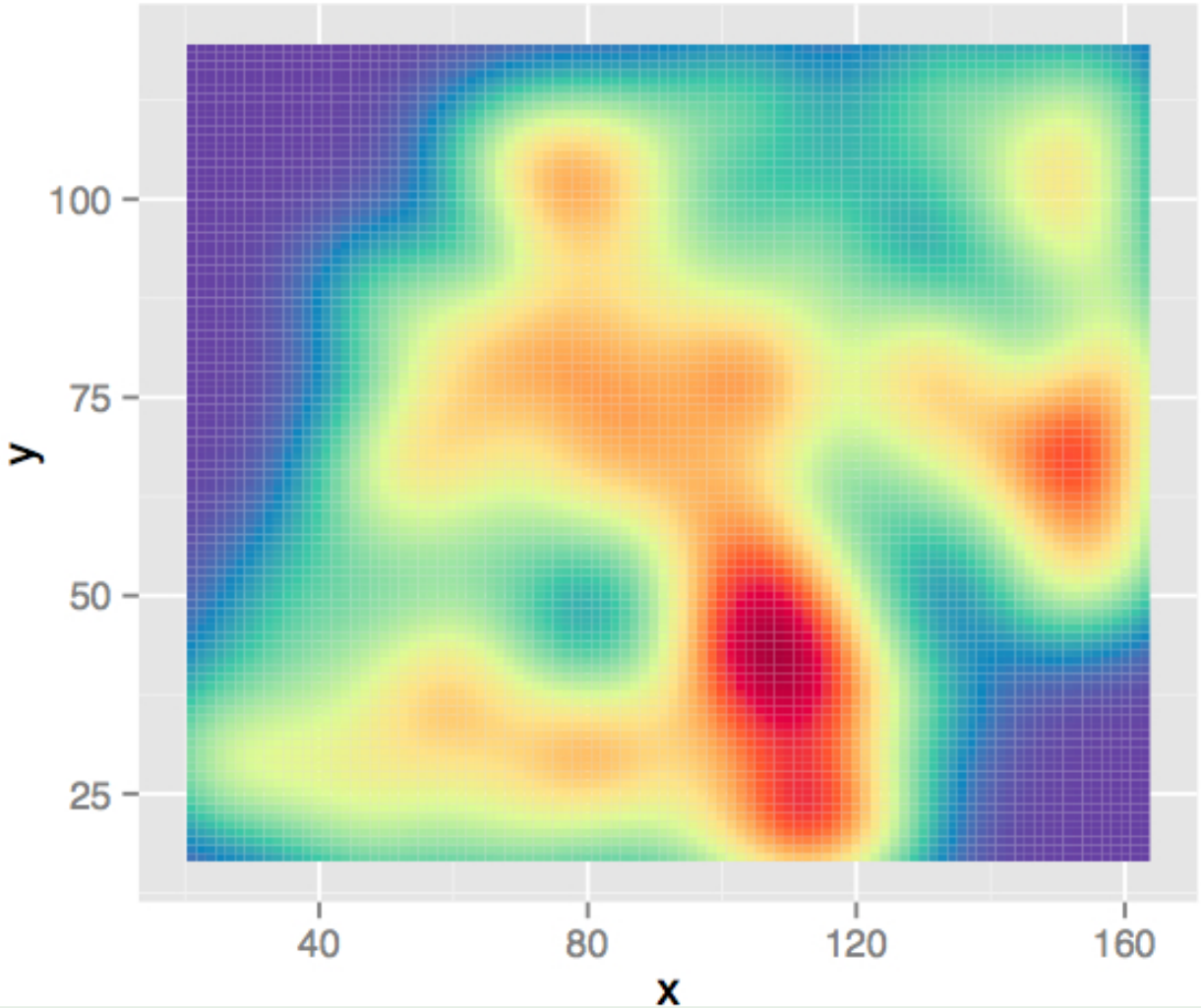
- *Discrete* means "countable" (can be listed in an order)
- We can count the number of animals.

So, Discrete Objects, Then?



- Discrete mathematics involves being able to count (*list*) things individually.

... And, Continuous Objects?



- "Overlapping" objects cannot be counted (i.e., listed) separately.

Non-*Discrete* and *Un-Countable* Objects

Really big amounts of things

- Are the numbers of grains *un*countable (i.e., unlistable)?
- Is anything *un*countable at the beach?
- How do we count an uncountable object? Why?

Relationships to Computing

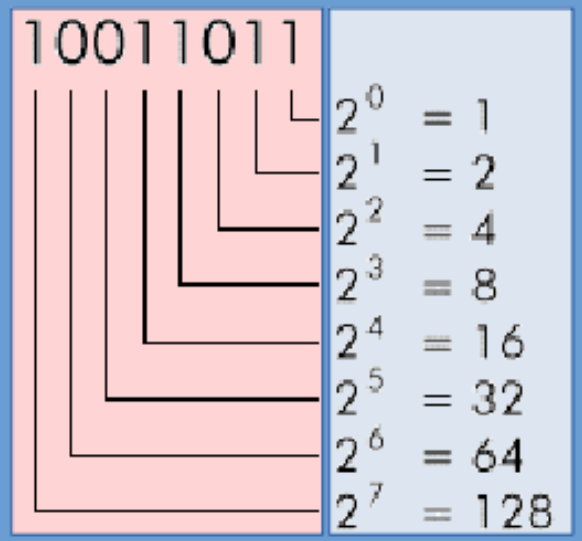
Computer MUST be able to count to compute

Binary Numbers

In mathematics and digital electronics, a binary number is a number expressed in the base-2 numeral system or binary numeral system, which uses only two symbols: typically, 0 (False, zero) and 1 (True, one).

- Computers use binary to function
- Processes (i.e., memory, computation, networking) are broken down into binary-driven procedures

Binary Numbers



Binary Value	Decimal Representation	Decimal Value
	8 4 2 1	
0 0 0 0	0 + 0 + 0 + 0	0
0 0 0 1	0 + 0 + 0 + 1	1
0 0 1 0	0 + 0 + 2 + 0	2
0 0 1 1	0 + 0 + 2 + 1	3
0 1 0 0	0 + 4 + 0 + 0	4
0 1 0 1	0 + 4 + 0 + 1	5
0 1 1 0	0 + 4 + 2 + 0	6
0 1 1 1	0 + 4 + 2 + 1	7
1 0 0 0	8 + 0 + 0 + 0	8
1 0 0 1	8 + 0 + 0 + 1	9
1 0 1 0	8 + 0 + 2 + 0	10

- Computing implies digital processing
- Computing binary values is a *countable* task.
- Can anything, or any number, that a computer computes be written in binary?

Countable and Not Countable?

What can be *listed* and what cannot be listed?

- Get into groups and discuss the following. Take notes to report back to the class.
- Can you think of **countable** objects?
- Can you think of **un-countable** objects?
  - Can you think of types of numbers that may fit into each of these above groups?