LECTURE STYLE COURSES

I. ITERATION

Enter a number in your calculator. Now press the square button repeatedly and watch the numbers evolve on the screen. Can you predict the long-term behavior of this sequence? Will the numbers get larger and larger, or does something else happen? What happens if you change the starting number? If these questions are too easy, are there any other operations you can try to perform repeatedly?

In this course we will study discrete dynamical systems. Instead of calculating on paper, we will use Python to perform numerical experiments, which will allow us to quickly formulate and test our conjectures. We will then use Desmos to gain insight into the causes of the behavior we are observing.

II. FINITE FIELDS AND THEIR APPLICATION

You may have spent many years working with number systems like the rational numbers (e.g. 3/4), the real numbers (e.g. 3.1415926...) and the complex numbers (e.g. 3 + 4i). In each of these number systems you can add, subtract, multiply and divide. But there are actually many other number systems out there in mathematical nature with these four operations. And as we can carry out the same kinds of operations in these systems, we can also ask the same kinds of questions. Unlike exotic systems such as the real numbers, many of the systems we study can be stored entirely on a computer - no approximations needed. These systems might seem like little novelties, but they exhibit patterns that have fascinated mathematicians for centuries, and have real-world applications as well.

In this course we will gain hands-on experience performing computations in such systems. Just as we do over the real numbers, we will factor polynomials and solve equations over such systems. And we will see applications of the theory we develop to geometry, combinatorics and error-correcting codes.

III. ANALYTIC COMBINATORICS

An active area of computer science and discrete math research is centered around the remarkable observation that we can use the idea of the *generating function* to transform many combinatorics problems into questions about algebra. For some problems - like Sicherman's Dice - we can turn a question about counting into a question about polynomials and factorization. Where things get really exciting, though, is where we take on infinite combinatorial classes and leave the world of polynomials for power series on the complex plane. In this class we'll learn a powerful toolkit that allows you to solve large classes of combinatorial problems

Prerequisites: *Calculus, particularly convergence and manipulation of power series. Combinatorics at the level of AoPS Intermediate Counting and Probability.*

IV. COMPUTER ORGANIZATION

Computers may seem complex but they are really just well organized systems of simple components. We'll start by building circuits to compute boolean functions and we'll discover how mathematical induction can shows how to build a circuit to compute any boolean function using copies of just one two-wire gate - a gate we can build with a transistor. With this in hand we will survey the von Neumann architecture at the heart of the modern computer and then take a peek at other ways we can make physics compute for us: with DNA, with billiard balls, and even with quantum mechanics.

V. COSMIC DISTANCE LADDER

Eratosthenes was able to measure the circumference of the earth. Aristarchus built on this to measure the distance to the moon, and even to the sun. He realized the sun was enormous and very far away, and concluded Earth can't be the center of the universe. Knowing the size of the earth's orbit, Bessel was able to use parallax to measure the distance to nearby stars. These stars were seen to have an interesting characteristic: their absolute brightness mostly depended on their color - this let us measure the distance to much more distant stars and even measure the size of our galaxy. The progression continues until we measure the size of the observable universe. Each of these steps builds on the previous one, and lets us climb a bit further up this "cosmic distance ladder". We'll climb as far as we can, learning subtle geometry, algebra, and perhaps some relativity. We'll also touch on primary sources and learn about the history of thought over the last 2500 years.

VI. GEOMETRY OF CURVATURE

Starting with some Geometry-by-experiment we learn the need for proof and axioms and build our own simple axiom systems. Not satisfied, we turn to Euclid and try our hands at a few proofs to see the power of this system that stood the test of time for 2000 years. But are we satisfied? Of course not. We turn back to experimental geometry and construct models that "break" Euclidean geometry. By understanding why Euclid's rules did not apply we will learn the significance of each of Euclid's axioms at a deep level and open the door to the geometry of curved surfaces and the remarkable properties of symmetric spaces.

VII. NUMBER REPRESENTATION

In mathematics and computer science research, it is often a breakthrough in how we represent a problem that leads to a cascade of breakthroughs in problem solving. In this class we're going to see how our technology for number representation has enabled - and limited! - our ability to do math over the last 5000 years. We'll see systems used by the Egyptians, Indians, Greeks, Romans, and Arabs. We'll study Archimedes' "Sand Reckoner" and see how he was able to explore numbers unfathomable to his predecessors. We will finish the class by revisiting one of the entrance exam problems and asking "What's the biggest number you can write with 40 characters?"

VIII. RECURRENCES, COUNTING, AND APPROXIMATION

LAB STYLE COURSES

I. COMPUTATIONAL LINGUISTICS PROBLEM SOLVING

Can you decode a Hittite tablet? Or piece together the meaning of a Quechua phrase from a few known fragments? How on earth could people figure out conclusively that Linear-B was used to write a previously unknown dialect of ancient Greek?

In this class we will explore language problems from the NACLO. These problems often seem impossible to start. We will see that through systematic and careful reasoning - in other words with logic - we can unlock new knowledge with confidence that intuition will never provide.

II. CURVATURE MODELS

All are welcome, but this lab will complement the "Geometry of Curvature" course. We will build beautiful models of hyperbolic, elliptic, and irregular spaces using paper, geofix, zometool, and maybe even sewing.

III. NUMBER THEORY WITH COMPUTERS

All are welcome, but this lab will complement the "Number Theory with a Computer" course, but give the freedom for deep dives and extra programming time.

IV. READING EUCLID IN GREEK

Learn enough of the structure and vocabulary of ancient Greek to start reading Euclid in Euclid's own words. We won't just learn language, we'll learn about scholia and how scholars kept discussion of Euclid alive for 2000 years. Plus we might make some convincing fake manuscripts.

This lab will have points of contact with "Geometry of Curvature", "The Cosmic Distance Ladder", and "Number Representation".

V. READING NEWTON IN LATIN

Learn enough about the structure and vocabulary of Latin to start reading and deciphering Newton's writing from his own hand. We'll decipher what his insights from his own words. Plus we'll write our own latin nuggets for future scholars to ponder.

VI. POLYHEDRA MODELS

Learn about the spherical symmetry groups and how they can be visualized through polyhedra. Along the way we'll see the structure of the families of Platonic and Archimedean solids, then explore duality through the Catalan solids.

Plus we'll build some really cool models with paper, zometool, and geofix.

VII. UNDERSTANDING POLYTOPES WITH SOFTWARE AND MODELS