CLASS DESCRIPTIONS—WEEK 3, MATHCAMP 2016

Contents

9:10 Classes

Advanced Topics in... Sorting? (𝅘𝅥𝅮𝅘𝅥𝅮, Zach, Tuesday–Saturday)
What seems at first a trivial algorithmic task—arranging a list of numbers/names/campers in order—is actually just the beginning of many beautiful, difficult, and even unsolved problems in theoretical computer science. This class is not a standard laundry list of classical sorting algorithms (bubble sort, merge sort, etc.), though we'll certainly find uses for these. Instead, we'll discuss more advanced, meatier questions about sorting or sorted data, such as: How can we search sorted data *faster* than binary search, and how does this speed up many geometric algorithms? (A topic near and dear to my research.) What gains can we get from many processors working together to sort the same data? What sorting methods are better for humans, even if they may be slower for computers? (This last one we'll investigate empirically, of course.)

This is a *theoretical* computer science class; no actual computer code will profane its purely mathematical discussions, nor is programming knowledge required or especially relevant.

Homework: Optional.

Prerequisites: Knowledge of binary search is recommended (it's easy to look up or ask me about), as is knowledge of any sorting algorithm, such as bubble sort (ditto). Familiarity with "Big-O notation" is helpful but not required.

Calculus on Young's Lattice. (𝅘𝅥𝅮𝅘𝅥𝅮𝅘𝅥𝅮, Kevin, Saturday)
Young's lattice is a beautiful poset capturing the structure of partitions, with connections to combinatorics, geometry, and representation theory. Young's lattice is almost unique among posets[1], with the remarkable property that we can do calculus on it. We'll discuss how, and we'll see how doing calculus on Young's lattice yields powerful enumerative results, including some famous identities involving Young tableaux. Plus, if you like exponentiating things in Assaf's class, we'll even get a chance to exponentiate xD!

Homework: None.

Prerequisites: None.

Domains and Factorization: When Everything Goes Wrong. (𝅘𝅥𝅮𝅘𝅥𝅮𝅘𝅥𝅮, *Alfonso Gracia-Saz*, Tuesday–Saturday)
You know that every integer can be written as a product of primes in a unique way. But, are you sure this is true? It turns out that proving the uniqueness part is not easy at all, even though we all take this fact for granted since kindergarden!

In this class you learn how to prove this rigorously, and you will also study other "number systems" where the same result fails. Sometimes the uniqueness part fails. Sometimes some numbers cannot be written as product of primes at all! Pathological examples are delicious.

This class will alternate between mini-lectures and time for you to work things out in class. The homework won't be long, but I will assume you are finishing it every day.

Homework: Required.

Prerequisites: Ring theory. Specifically, you need to be comfortable with the concepts of ring, ideal, and "ideal generated by."

Required for: Algebraic Number Theory (W4)

Cluster: Rings and Fields.

[1]Morally, there's only one other such poset, and no other lattices! Conjecturally, at least.

Epsilons and Deltas. (🐦🐦, Sam, Friday)
In the early 18th century, Bishop Berkeley was fed up with mathematicians calling religion irrational, and wrote the wonderfully titled *The Analyst, or a Discourse Addressed to an Infidel Mathematician.* Here he ripped into the philosophical foundations of Newton's and Leibniz's calculus. Several mathematician's then tried to respond by putting calculus on a rigorous foundation, culminating in Cauchy's epsilon-delta formalization. More than that, Cauchy was a wonderful and colorful mathematician. He taught engineers at the Ecole Polytechnique, where he was notorious for forcing them to learn modern analysis instead of computational methods; he basically wrote the book on analysis; and he stuck to his principles (rightly or wrongly) at all costs.

TL;DR: in my book, Cauchy is the quirky hero who saved calculus. Come to this class to find out why!

Homework: None.

Prerequisites: Basic calculus, including the formal epsilon-delta definition of a limit and limit definition of a derivative will help.

King Chicken Theorems. (🐦, Marisa, Friday)
Chickens are incredibly cruel creatures. Whenever you put a bunch of them together, they will form a pecking order. Perhaps "order" is an exaggeration: the chickens will go around pecking whichever chickens they deem to be weaker than themselves. Imagine you're a farmer, and you're mapping out the behavior of your chickens. You would like to assign blame to the meanest chicken. Is it always possible to pick out the meanest chicken? Can there be two equally mean chickens? Are there pecking orders in which all the chickens are equally mean?

Homework: Optional.

Prerequisites: None.

Natural Language Understanding. (🐦, *Greg Burnham*, Saturday)
Human language is tantalizingly close to being a formal system. We certainly *feel* like there is a clear relationship between the words we express and the facts these words convey. If this intuition were true, however, then we should be able to write computer programs to perform linguistic tasks (e.g., read a document and answer questions about it). But we've been trying to write such programs for 50 years, and the results are mixed at best.

This class will be a quick survey of some interesting topics in the (very broad) field of computational natural language understanding. We'll try to motivate why it's so difficult to write computer programs capable of performing linguistic tasks and then describe what tasks computers *can* currently perform, focusing on how recent algorithmic and technological progress has allowed for improved performance. We will conclude by noting that the big problems remain unsolved and speculating on what might be necessary for the next steps forward.

As a teaser, here is my favorite short example motivating why computational language understanding is hard. Consider the following two sentences, which differ only in the last word:

- "The cat caught the mouse because it was clever."
- "The cat caught the mouse because it was careless."

Now, what does the pronoun "it" refer to in each sentence? Humans share a clear intuition about the right answer to this question. And yet, consider what it would take to write a computer program with this same capability. That's the problem in a nutshell.

Homework: Recommended.

Prerequisites: None.

Nonzero-Sum Games. (🏃, Pesto, Tuesday–Thursday)
Sarah wants to take the Combinatorial Games class and Stephanie wants to take the Board Games class, but they conflict. Even more than either of them wants to take their preferred class, though, they want to be in the same class, to confuse their teachers. Games like combinatorial games and board games, with only one winner, have optimal strategies, although they may be hard to find. In Sarah and Stephanie's class-selection game, it's not clear what an optimal strategy means. We'll talk about and play games where everyone could win, or everyone could lose, or anything in between. We'll also discuss the game-theoretic implications of the golden rule, threats to shut down the government, and cloning.
Homework: Recommended.
Prerequisites: None.
Cluster: Games Mathematicians Play.

Pythagorean Triples, Diophantine Equations and Fermat's Last Theorem. (🏃🏃→🏃🏃🏃, *John Mackey*, Tuesday–Saturday)
Why should anyone spend time showing that the sum of the cubes of two positive integers is never the cube of an integer? Is it simply that we all enjoy a good challenge or want to be famous? Perhaps it is a case of the journey being more important than the destination.

Please consider joining us for the journey along topics related to Fermat's Last Theorem. We'll discuss Pythagorean Triples, and explore different ways of classifying them, before moving on to Fermat's problem and related variants. Along the way, we will encounter different types of math, and witness the power of co-mingling ideas.
Homework: Recommended.
Prerequisites: Induction, elementary proofs. Knowing modular arithmetic and the Euclidean algorithm would be useful, but is not strictly necessary.

Wallis and His Product. (🏃🏃→🏃→🏃🏃🏃🏃, *Jon Tannenhauser*, Tuesday–Thursday)
John Wallis (1616–1703) published what is essentially the infinite product formula

$$\frac{\pi}{2} = \frac{2}{1}\cdot\frac{2}{3}\cdot\frac{4}{3}\cdot\frac{4}{5}\cdot\frac{6}{5}\cdot\frac{6}{7}\cdot\frac{8}{7}\cdot\frac{8}{9}\cdots$$

in his 1656 treatise *Arithmetica infinitorum*. His work mixed suggestive analogy, intuitive leaps, and sheer moxie—and sparked reactions ranging from awe to skepticism to spluttering rage. We'll trace Wallis's argument (day 1; 🏃🏃) and discuss how and why it touched off a battle in a long-running 17^{th}-century war over infinitesimals, which was actually—although no one knew it at the time—a struggle over the foundations of calculus (day 2; 🏃). Then we'll look at a recent (October 2015) derivation of the Wallis product via the quantum mechanics of the hydrogen atom (day 3; 🏃🏃🏃🏃)!
Homework: Recommended.
Prerequisites: None.

10:10 Classes

Field Extensions and Galois Theory (Week 2 of 2). (🐾🐾🐾, Mark, Wednesday–Saturday)
This is a continuation of the week 2 class. If you would like to join and you're not sure what has been covered, check with Mark (and/or with someone who has been taking the class and who has good notes).
Homework: Recommended.
Prerequisites: Week 1 of Field Extensions and Galois Theory.
Cluster: Rings and Fields.

Graph Colorings. (🐾🐾, Mia, Wednesday–Friday)
Normal graph colorings are *so* cliche and old-school. Like, 1800s old-school. If you want to be modern, like bell bottoms, disco, and pet rocks of the 1970's, you might like list coloring. List coloring is a system in which each vertex comes with a list of acceptable colors and you may only color a vertex using a color from its list.

Or, if you prefer Rainbow Brite, Hacky-Sack, and John Hughes movies about angsty teenagers of the 1980s, you'll be wiggin' out over fractional coloring. Fractional coloring allows the graph theorist to color each vertex with multiple colors: for example, I could paint a certain vertex half red, a quarter blue, and a quarter pink.

And with all these new, rad coloring schemes, we get new, rad chromatic numbers. Will these chromatic numbers differ from our old-school χG? Can they differ by arbitrarily much? In this class we will investigate these questions and see some gnarly graphs.
Homework: Recommended.
Prerequisites: Introduction to Graph Theory.
Cluster: Maps, Graphs, Colors, Walks.

Problem Solving: Symmetry, Parity, and Invariants. (🐾🐾🐾, *Joshua Zucker*, Wednesday–Saturday)
Parity is the simple idea that numbers are either odd or even. It's an example of an invariant, which is something that doesn't change when you do certain operations: the parity doesn't change if you only add and subtract even numbers from your starting number. Another invariant is the total of a list of numbers: if you rearrange the numbers, or even if you add up some pairs of numbers, the total stays the same. This is because addition has symmetry: the order of the inputs doesn't matter.

In this class we will look at a variety of operations and discover the invariants and symmetry that they have, and apply those to solve some easy problems, some hard problems, and some seemingly impossible problems. What happens if you combine numbers with the rule $xy+x+y$ instead of simple addition? How can you organize cups to be flipped right side up on a spinning table you can't even see?
Homework: Optional.
Prerequisites: None.
Cluster: Problem Solving.

The Hadwiger–Nelson Problem. (🐌🐌, Riley, Saturday)
Hadwiger and Nelson (independently) posed the following problem: Take the Euclidean plane and connect points with an edge exactly when they are at distance 1 from each other. What is the chromatic number of the resulting graph?

Currently, no answer is known. Better yet, there is some evidence that the number depends on the axioms of set theory.

This class will state what is known about the Hadwiger–Nelson problem and build some intuition about strange infinite graphs.

Homework: Recommended.

Prerequisites: Introduction to Graph Theory.

Cluster: Maps, Graphs, Colors, Walks.

The Topology and Geometry of Surfaces. (🐌, Jane, Wednesday–Saturday)
A Square lives on a surface called Flatland, a two-dimensional world. For many years, most Flatlanders assumed that their world was a giant plane, extending infinitely in all directions. However, the curious and adventurous A Square wasn't satisfied by this assumption and wanted to test it. One day, he set out in a straight line and came back to his starting point, without changing direction. Another day, he took a long walk and came back to his starting position to find that his whole world was now mirrored. Certainly, A Square's experiments proved that Flatland wasn't an infinite plane, but what was it?

In this course, we'll explore the question of how to tell different surfaces (two-dimensional spaces that locally look like planes) apart with both topological and geometric tools. We'll take a hands-on approach to learn about what different surfaces there are and what geometries they can have. We'll also use this knowledge to help us understand the geometry and topology of three-dimensional spaces, and speculate as to the shape of our universe.

Homework: Recommended.

Prerequisites: None.

Cluster: The Shape of Things.

What Can We Exponentiate? (🐌🐌🐌, Assaf, Wednesday–Saturday)
e^x is **the best** function. It's always defined, it's always continuous, always differentiable, and generally awesome. e^x can make any matrix invertible. e^x can make vector spaces into manifolds. e^x can turn anything into a group. e^x can solve differential equations. e^x can even travel through time itself.

All of this is true, and we will prove it all.

Homework: Recommended.

Prerequisites: Linear Algebra (Weeks 1 and 2).

Cluster: Summing Series.

11:10 Classes

A Tale of Combs and Hedgehogs. (🐌🐌🐌🐌, Alfonso + Chris, Tuesday–Saturday)
Once upon a time a princess was walking through the woods and came across a hedgehog. "I'm a beautiful prince," the hedgehog said, "all you need to do to free me from my curse is comb me nice and smooth." The princess was of course well-educated in mathematics, paid him no mind and moved on, for she knew that such a task was impossible and that he would be cursed forever.

Why was the princess so sure to move on? Because the hairy ball theorem states that no ball can be combed in such a way that no hair sticks up, and that there's no parting in the hair. In this class

you will develop the proof for the hairy ball theorem on your own and on the way you will discover many important tools from algebraic topology!

This is a superclass that meets for two class hours a day (and possibly the first hour of TAU if we need it). You will be doing most of the work: we will provide worksheets with the right definitions and questions; you will spent a big chunk of the time working, alone or in groups, sometimes with our help, on all the steps of the proof. Some of the class time will be spent on presentation and discussion of your work.

This course is time-consuming, but all the work (homework included) is contained in the two to three daily hours.

Homework: Required.

Prerequisites: Some notion of continuity, basic group theory and linear algebra (images, kernels, and quotients).

Cluster: Algebraic Topology.

Projective Geometry. (𝆑𝆑, Sachi, Tuesday–Saturday)
When you look out along a pair of railroad tracks, they appear to meet at a point in the distance. Have you ever wondered what would happen if you decided to define such a point at infinity, where each pair of parallel lines met?

Sometimes I get sad thinking of all of those lonely parallel lines living their lonely Euclidean existence just passing each other by and never getting to say hi.

In fact, there is a happier universe out there, where all parallel lines meet at infinity, and in this universe, many interesting things happen. We can see the behavior of hyperbolas, parabolas, and cubics from the "other side", at infinity. We will see that for many of purposes, this is the right geometry to consider, and it makes many types of computation and visualization easier.

Homework: Recommended.

Prerequisites: None.

Random Graphs. (𝆑𝆑𝆑, Misha, Tuesday–Saturday)
Imagine taking a graph on 1000 vertices with no edges (a very boring object), and start putting down edges with randomly chosen endpoints one at a time. (You should avoid picking the same edge twice, although in the first few thousand steps this won't affect the result very much at all.)

Can we predict approximately how many edges the graph will have before it becomes connected? Before it contains your favorite subgraph? How will its chromatic number change with the number of edges?

In this class, we will study the evolution of the random graph, a beautiful story that will incidentally provide us with limitless examples and counterexamples for all sorts of problems in graph theory. Along the way, we will develop a number of techniques of the probabilistic method.

Homework: Recommended.

Prerequisites: Graph theory. (I will assume familiarity with graph terminology, subgraphs, trees, chromatic number. If unsure, speak with me at ϵ-TAU.)

Cluster: Techniques in Graph Theory.

Representation Theory of Finite Groups (Week 1 of 2). (𝆑𝆑𝆑𝆑, Mark, Tuesday–Saturday)
It turns out that you can learn a lot about a group by studying homomorphisms from it to groups of linear transformations (if you prefer, groups of matrices). Such a homomorphism is called a *representation* of the group; representations of groups have been used widely in areas ranging from quantum chemistry and particle physics to the famous classification of all finite simple groups. For example,

Burnside, who was one of the pioneers in this area along with Frobenius and Schur, used representation theory to show that the order of any finite simple group that is not cyclic must have at least three distinct prime factors. (The smallest example of such a group, the alternating group A_5 of order $60 = 2^2 \cdot 3 \cdot 5$, is important in understanding the unsolvability of quintic equations by radicals.) We may not get that far, but you'll definitely see some unexpected, beautiful, and important facts about finite groups in this class, along with proofs of most or all of them. With any luck, the first week of the class will get you to the point of understanding character tables, which are relatively small, square tables of numbers that encode *all* the information about the representations of particular finite groups; these results are quite elegant and very worthwhile, even if you go no further. In the second week, the chili level may ramp up a bit (from about $\pi + \frac{1}{2}$ to 4) as we start introducing techniques from elsewhere in algebra (such as algebraic integers, tensor products, and possibly modules) to get more sophisticated information.

Homework: Recommended.

Prerequisites: Linear algebra, group theory, and general comfort with abstraction.

Cluster: Groups.

dCalculus. (🌶, Jeff, Tuesday–Saturday)
Can you guess the next number in the sequence?

$$1, 3, 6, 10, 15, ...$$
$$1, 1, 2, 3, 5, 8, ...$$
$$1, 2, 4, 8, 16, 32, ...$$
$$1, -1, 1, -1, 1, ...$$

I bet you solved all of these by recognizing that they were the solutions to discrete differential equations!

We'll take tools from standard single variable calculus and replace them with their discrete relatives, developing dDerivatives, dIntegrals, dFTOC, dTaylor Series and the dFourier transform. Then, we'll see what dProblems we can dSolve with our dTools, including why e is almost 2, and the hyperplane division problem.

Homework: Required.

Prerequisites: None.

Cluster: Summing Series.

1:10 Classes

A Crash Course in Axiomatic Probability. (🌶🌶🌶, Sam, Tuesday–Saturday)
This class will be a brisk walk through some of the most fundamental topics in probability. We will start from Kolmogorov's axioms and build our way through formal notions of independence, conditional probability, and random variables. By the end of the course, we'll have sufficient tools to prove some high-level theory and asymptotic results! We will mostly focus on discrete probability—so if you havn't seen much calculus that's fine—but will touch briefly on continuous random variables.

Throughout the class, we'll look at a few fun "applications" of probability. Typically, these will be applications to other areas of mathematics (like Graph Theory!).

Homework: Required.

Prerequisites: You should know the following words and corresponding symbols: (finite and countable) union, intersection, complement, and partition. Calculus will help for 20 minutes of this course. We'll also do one or two examples from graph theory, where knowing very basic terminology will be helpful

(edge, vertex, clique, independent set, complete graph), but these are just fun asides and can be safely ignored.

Finitely-Generated Algebras. (𝒿𝒿, Susan, Tuesday–Saturday)
Have you ever wanted to build an algebraic structure that does something *really* terrible? Sure, noncommutativity is cool and all, but how would you define a ring that has an element with a right inverse but no left inverse? Or maybe two nonzero elements a and b such that $ab = 0$, but ba is nonzero? In this class, we'll be talking about how to build rings with specific bad behaviors using polynomials as our building blocks. We'll talk about how to tell when two elements of these rings are "the same", and how we can measure the size of the objects we've built.

Homework: Recommended.

Prerequisites: I'll use the word "ring", so knowing the definition might be helpful, but probably not absolutely necessary.

Cluster: Algebraic Novelties.

Functional Programming. (𝒿𝒿, Nic, Tuesday–Saturday)
Functional programming languages are ones that encourage the programmer to focus not on which instructions a program is supposed to follow but on what the output is supposed to be. When writing in a functional style, variables can't be modified, data structures can't change their shape, and functions can't do anything with their input but use it to compute and return an output.

This might sound like coding with both hands tied behind your back, but once you get used to it, the functional paradigm can be incredibly powerful. By giving up side-effects, you gain the ability to reason more easily about what your program is doing. For example, here's a function which takes a binary tree with entries at each leaf and a function taking those entries to integers, then applies the function to all those entries and sums the results:

```
let rec sum_tree f t = match t with
| Leaf n -> f n
| Branch (l, r) -> sum_tree f l + sum_tree f r
```

In this class, we'll explore functional programming through the language OCaml. We'll meet for two hours a day, and most class time will be spent writing code to solve problems that will be provided. The problems will be on lots of different topics, and several will be built around writing your own interpreter for a simple programming language. You'll be encouraged to solve problems at your own pace; there should be more than enough to keep you occupied for the whole week, but if you finish I'll come up with more for you to do.

Homework: Recommended.

Prerequisites: Some programming experience, but not necessarily in a functional language.

Cluster: Mathematics and its Applications.

Marathons

Algebraic Groups. (𝒿𝒿𝒿𝒿, Don, Tuesday–Saturday)
When algebraists first made serious attempts to study $GL_n(\mathbb{C})$, the group of invertible $n \times n$ matrices, they were struck by just how much algebra is actually taking place. First, it has a group structure. Second, it's a group of matrices, so it is central to linear algebra. Third, its definition depends on a ring, $\mathbb{C}$, in a way that is almost modular—you could replace it with another ring, and you'd have a new, somehow related group to study.

And so, using a bit of Galois theory and category theory that we'll develop along the way, that's exactly what they did! Instead of studying $GL_n(\mathbb{C})$, they studied GL_n, the abstract concept of a group of invertible $n \times n$ matrices—and other concepts like it.

In this marathon class (meaning that we'll meet for all but one class block, plus half of TAU, each day), we'll define algebraic groups, prove that every algebraic group is a subgroup of some GL_n, and ultimately classify semisimple algebraic groups, in a way intrinsically linked to the classification of finite simple groups.

Homework: Required.

Prerequisites: Group Theory, Ring Theory, Linear Algebra, Planning Meeting.

Cluster: Groups.

Colloquia

Ron Graham's Sequence. (*Joshua Zucker*, Tuesday)

It's hard to write about Ron Graham's sequence without giving too many spoilers. The surprises are a big part of the fun! So you may prefer to stop reading now.

Can you make a list of all the non-prime numbers? Sure, $1, 4, 6, 8, 9, \ldots$, no problem. OK, quick, what's the thousandth number in that list? Millionth? I wonder if there's an order we can put them in such that we could reasonably quickly find out the thousandth or the millionth number in the list, without having to find all the numbers before them. Well, Ron Graham found one! We'll see what his discovery was and how to prove that the sequence contains every non-prime number exactly once.

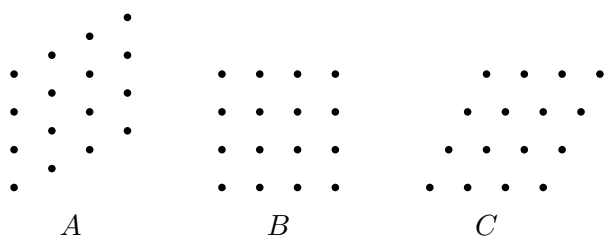
Crossing Numbers of Graphs: Origins and Recent Progress. (*John Mackey*, Wednesday)

When Paul Turan was interned in a Nazi labor camp during World War II he was assigned to a job which required that bricks be moved from kilns to storage depots. There were paths from each kiln to each storage depot, but the carts on which the bricks were moved had a tendency to overturn when crossing the path of other carts. Turan devised a system of paths which seemed to minimize the number of such crossings, but could not prove that his configuration was indeed best possible.

Zarankiewicz published a paper containing a flawed proof of the optimality of Turan's construction in the 1950s. Richard Guy exposed the flaw and introduced several generalizations of Turan's problem in 1969. All of the large conjectures from Guy's paper remain unsolved to this day, but recent progress (including some by CMU undergraduates) indicates that solutions may be near.

Lattices, Knots and Chaos. (Jeff, Thursday)

Here are a couple of different grids of dots:



All of them give examples of lattices: discrete $\mathbb{Z}\times\mathbb{Z}$ subgroups of $\mathbb{R}^2$. These beasties make themselves present in number theory, algebraic geometry, topology—they're a piece of mathematics that shows

up over and over again. However, it's not immediately clear how we can classify lattices; for instance, the first and third lattice in the above figure can be related to each other by a rotation.

We'll construct a topological space whose points parameterize lattices, and show how we can better understand operations on lattices through the geometry of this space. Then, we'll flip the relationship around, and restate a problem of the Lorenz attractor in terms of the geometry of lattices.

The Littlewood–Offord Problem. (Susan, Friday)

Don is going to offer a small group of campers a fabulous investment opportunity. One lucky camper collective will have the opportunity to invest 100 Suzin Bux in the 5/3 Bank of MCSP, at twice the usual interest rate! Here's the catch: this camper collective must have exactly 100 Suzin Bux between them.

To complicate matters, Vivian has decided that it would be fun if there were as much competition as possible for this fabulous investment opportunity. So she has been running around handing out extra Suzin Bux to campers, with the goal of creating as many 100-Suzin-Buk camper collectives as possible.

In this talk we will discuss the optimal strategy for Vivian. Note: this is not an endorsement of the Suzin Buk money system. Suzin Bux are not worth anything, and Susan has absolutely nothing to do with them.

Visitor Bios

Alfonso Gracia-Saz. Alfonso is an Assistant Teaching Professor at the University of Toronto. He has taught math in many places, including San Quentin State Prison in California, but Mathcamp remains his favourite. He is a fan of Percy Jackson, he sees algebraic topology in square dancing, his favourite board game is the Castles of Burgundy, and he makes a mean paella.

Greg Burnham. I'm Greg. I was a camper in '04–'06 and a JC in '07, '08, and '10. I majored in math in college, with the intention of going to grad school, but got cold feet my senior year and instead took the first job I thought sounded interesting. I've had a few jobs since then. Currently, I work at a small AI lab, doing research in the broad area of language understanding. I live in Brooklyn, NY with my girlfriend Courtney, who may also be around camp when I am.

John Mackey. John Mackey struggled in high school and attended Kent State University, which was required to take all graduates of Ohio public schools. He was delighted to find interesting people and wonderful teachers at Kent State, who inspired him to learn and grow. Upon graduation he went to the University of Hawaii to learn more about Polynesian and Asian culture, as well as to learn how to surf and to develop a deeper understanding of math.

After completing his Ph.D., he became a John Wesley Young Research Instructor at Dartmouth College, where he taught from 1996–2000. He moved on to be a Preceptor of math at Harvard from 2000–2003. Needing bigger challenges and cheaper real estate, he moved to Pittsburgh, PA, to direct undergraduate studies in mathematics at Carnegie Mellon University.

He won the William H. and Francis S. Ryan Award for Meritorious Teaching in 2013 and was a Co-PI on the DEBT-M Project, which is devoted to closing the opportunity gap for marginalized secondary mathematics students in Pittsburgh.

Jon Tennenhauser. Jonathan Tannenhauser teaches math at Wellesley College. In the past he has worked in theoretical physics, finance, and the genomics of birdsong, and more recently he has become interested in the history of mathematics. This is his sixth visit to Mathcamp.

Joshua Zucker. Joshua discovered his love of number theory at the Ross Program. He has become a freelance math teacher. He was the founding director of the Julia Robinson Mathematics Festivals, has directed the Bay Area Math Olympiad, helped run the first Math Teachers' Circle, and has taught many classes for Art of Problem Solving. He is also a regular participant in the US Puzzle Championship and was a member of the US Sudoku Team at the World Sudoku Championships in Croatia.