Categories and Programming
IAP 2020 – 18.S097

In this course we explain how category theory—a branch of mathematics known for its ability to organize the key abstractions that structure much of the mathematical universe—has become useful for writing elegant and maintainable code. In particular, we'll use examples from the Haskell programming language to motivate category-theoretic constructs, and then explain these constructs from a more abstract and inclusive viewpoint. Hands-on programming exercises will be used to demonstrate categorical ideas like "the universal property of products" in working Haskell code. A rough list of topics includes:

1.   Sets, types, categories, functors, natural transformations
2.   Universal constructions and associated data types
3.   Adjunctions and cartesian closed categories
4.   Algebras, catamorphisms, anamorphisms
5.   Monads, comonads, Kleisli arrows
6.   Monoids, monoidal categories, lax monoidal functors, applicatives
7.   Profunctors, (co)ends, optics

We will assume no background knowledge on behalf of the student, starting from scratch on both the programming and mathematics.

Dates: January 07 – 31, 2020
Time: 2 – 3 pm, MTWRF, plus discussion hour 3 – 4pm.
Location: Room 4-163.
Instructors: David Spivak, Bartosz Milewski, and Brendan Fong
Credit/audit: For credit (3 units), sign up for 18.S097. You are also invited to audit.
URL: http://brendanfong.com/programmingcats/