



Patterns in Programming

Prof. Jeremy Gibbons

Computing at the University of Oxford



+ Tuesday, March 24, 2026 10:00
Auditorium 106 at IIS new Building +

Abstract

Many programs that are apparently very different on the surface turn out to be very similar underneath: they are varying instances of a common pattern. Sometimes we can capture the pattern as a generic program scheme, and obtain the instances by instantiating parameters of the scheme. This is a beneficial situation to be in, since any effort spent in understanding, refining, debugging, or verifying the pattern can be exploited for all of the instances.

One particular such situation arises from the well known insight that program structure often follows data structure. In that case, the generic program scheme is a pattern of computation induced by the shape of the data it manipulates, and the parameter is the datatype. Each instantiation to a specific data structure induces a corresponding program structure instance. The program schemes and parameters typically arise from standard algebraic and order-theoretic concepts such as monoids, semirings, and preorders, and more generally from constructions in category theory.

I have just spent a sabbatical year writing a book “Patterns in Functional Programming” about this phenomenon, to be published by Cambridge University Press later in 2026. The book is expressed using functional programming, but the ideas apply to all of programming. I will describe the premise of the book, and present some pretty examples.

Biography

Jeremy Gibbons is Professor of Computing at the University of Oxford, where he leads the Algebra of Programming research group, and is a lecturer on the part-time professional postgraduate master's programme in Software Engineering. His research interests are in programming languages, especially functional programming and patterns in programming. He is Editor-in-Chief of the journal *The Art, Science, and Engineering of Programming*, an Editor (and until recently Editor-in-Chief) of the *Journal of Functional Programming*, an Editor of *Compositionality*, a member (and former chair and secretary) of IFIP Working Group 2.1 on Algorithmic Languages and Calculi, a member of IFIP Working Group 2.11 on Program Generation, and former Vice Chair of ACM SIGPLAN.

For more information:
<http://www.iis.sinica.edu.tw/>

