In typed functional programming there is often the need for combining pure and monadic (or other effectful) computations, but the required lifting must be done manually by the programmer and may result in cluttered code. This thesis explores ways to allow the compiler to perform this task automatically. Several possible approaches are described, where the final one reduces the task to solving a system of linear diophantine equations. Apart from monads, the described method is also considered for the case of applicative functors as another abstraction to represent effectful operations.