From Mathematical Functions to Syntax

1 LCF Considered as a Programming Language

```
@article{gdp:lcf-as-pl,
  author={Plotkin, G. D.},
  title={LCF Considered as a Programming Language},
  journal={Theoretical Computer Science},
  pages={223--255},
  year=1977
}
```

Summary: Plotkin investigates the relation between the denotational and the operational semantics of a programming language. As a working example, he picks PCF a language based on Scott's Logic of Computable Functions. He demonstrates that even though denotational and operational semantics are in agreement when it comes to how they map PCF programs to their results, elements of the denotational space cannot be expressed as PCF function definitions. Thus the equality of functions in the denotational space is different to contextual equivalence.

Evaluation: This is a seminal paper. It is the first result on the relation between denotational and syntactic approaches to the semantics of programming languages. Moreover, it introduces the two fundamental properties that determine this relation: adequacy and full abstraction. Establishing the latter remained an open problem for PCF for more than a decade leading to a significant body of results including the invention of game semantics. Finally PCF itself has become an established and popular model of programming languages.