Putting Syntax to Work, for Real

1 Call-by-Name, Call-by-Value, and the λ-calculus

```
@article{gdp:cbn-cbv,
  author={Plotkin, G. D.},
  title={Call-by-Name, Call-by-Value, and the λ-calculus},
  journal={Theoretical Computer Science},
  pages={125--159},
  year=1975
}
```

Summary: Plotkin investigates the relation between ISWIM and the λ-calculus. In particular, he tries to determine what version of the λ-calculus provides an equational theory for ISWIM programs. He establishes that the call-by-value λ-calculus models faithfully the meaning of ISWIM programs as evaluated by an SECD machine and the equational theory of that version of the λ-calculus implies contextual equivalence for ISWIM. In addition, he shows that the call-by-value λ-calculus models faithfully a call by name version of ISWIM. Finally, he studies the relation between call-by-value and call-by-name ISWIM via their corresponding calculi. He demonstrates that one can simulate the other with two simulations that map terms of one calculus to terms of the other. The mappings preserve equational reasoning but not contextual equivalence.

Evaluation: This is a seminal paper. It is the starting point of operational semantics and demonstrates that they are an extremely useful method to express the meaning of programs and prove properties of programming languages. Thus it kick-started a thread of work that established operational semantics as the main tool for studying programming languages. Furthermore, the limitations of the framing of the problem (ISWIM without mutation and control operators) and the negative results (the translation between the call-by-name and call-by-value calculi is incomplete with respect to contextual equivalence) lead to a series of questions that researchers have been studying since.