The Meaning of Programs (Semantics)

Luca ACETO

Formal Languages { Programming Languages
Specification Languages (UML)

Languages { Syntax : BNF (context-free grammars)
Semantics : classically "textually" specified { ambiguous
contradictory

Formal descriptions of the semantics can be ambiguous, BUT we can check if this is the case.

Formal semantics to develop a better (informal) mental picture of the semantics of your language.

Formal semantics of REAL programming languages ARE DONE.

Formal semantics CAN, and SHOULD, avoid any reference to any precise implementation.

Formal semantics provides a framework to reason about programs : correctness, equivalence of (fragments of) programs.

Implementation : rapid prototyping, optimization, sophisticated "bright" implementation "tricks" (which can be FORMALLY proved SOUND)

Semantics { Operational : execution of the program in an "abstract machine"
Denotational : abstract values capturing the "effect" of programs
Axiomatic : logical properties afforded by the program

Complementary use : which framework is the most adequate in each circumstances ; "equivalences" between different definitions (it would be more adequate to talk about consistency here, because it is our intention to specify different things, depending on the kind of used semantics).