

What is a *metavariable*?

A variable that ranges over the *metalanguage* (the notation of a description), not the *object language* (the language being defined).

What is a *metatheory*?

"The collection of true statements that we can make about some particular logical system (or programming language)--and, by extension, the study of such statements."

What's the difference between a *term* and an *expression*.

Expressions are a superset of terms. Terms are for syntactic phrases representing computations, while expressions in general may refer to other syntactic phrases too.

Why can the definition of a language's syntax get away with not mentioning parentheses?

The definitions actually describe trees. Parens are simply a way of resolving ambiguities when a term is represented a string.

Name four ways to define the syntax of a language.

- Using a grammar, in BNF form for example.
- By defining terms inductively.
- Using inference rules.
- By defining a procedure for generating all possible phrases.

Inference rules with no premises are called ...

... *axioms*.

Why are inference rules actually *rule schemas*?

They may reference metavariables. Hence each may be instantiated to any of an infinite set of *concrete rules* containing phrases in place of metavariables.

What are the three methods of inducting on terms?

- Inducting on the depth of terms.
- Inducting on the size of terms.
- Inducting on the structures of terms (on all possible immediate subterms).

What is a language's *semantics*?

The precise rules of how terms are evaluated.

What are the three primary approaches to formalizing a language's semantics?

- Operational semantics.
- Denotational semantics.
- Axiomatic semantics.

Define *operational semantics*.

An approach to formalizing semantics that specifies the behavior of a programming language by defining a simple abstract machine for it. The abstract machine uses the terms of the language itself (or higher level structures) as states and defines a transition function between states. The meaning of a term is the final state reached when starting with that term as the initial state.

In the framework of operational semantics,
proofs on the behavior of a program amount
to what?

Proofs of the correctness of the implementation of the language.

Define *denotational semantics*.

An approach to formalizing semantics that takes meanings of terms to be mathematical objects.

Giving a language denotational semantics
involves what?

Finding a collection of *semantic domains* and then defining an *interpretation function* mapping terms into elements of these domains.

What is *domain theory*?

The search for appropriate semantic domains for modeling various language features.

Define *axiomatic semantics*.

An approach to formalizing semantics that takes laws themselves as the definition of the language. The meaning of a term is just what can be proved about it.

What are the most difficult cases for denotational and axiomatic semantics.

- Denotational semantics: nondeterminism and concurrency
- Axiomatic semantics: procedures