

Ask me anything

0 questions

0 upvotes

Why is the abstract syntax for a language generally much simpler than the concrete syntax?

because it describes the syntax in less detail

because in abstract syntax we can assume that it can be parsed without having to get into technical details

Abstract Syntax eliminates details that are not interesting after the code has been parsed

Concrete needs to parse the language, abstract only handles the AST

In AST there are some assumptions (for example mathematical details as operator prioritizations)

Concrete should be readable more or less directly to a compiler while abstract is rather human readable

What's the difference between operational semantics and denotational semantics?

operational semantics is written for an abstract machine, where denotational semantics is more mathematical

In denotational semantics the program is mapped to mathematical objects while in operational semantics it is mapped to abstract machine

denotational tries to express things in mathematical expressions while operational semantics express things such that they could be interpreted with an abstract machine (like turing machine)

Operational: easy to implement but hard to reason about
Denotational with mathematical denotation, easy to reason about but hard to find solution

Why do we need separate semantic functions for each syntactic category (e.g., programs, statements, expressions etc.)?

Maybe because the different constructs have completely different indications on a hardware level

different categories have different rules -> they need other functions

test

Expressions always can be reduced to a value, where statements cannot

Why are semantic functions typically higher-order?

A higher order function is a function that takes a function as an argument, or returns a function?

Thanks, I got it now. :-) Btw, it's in Lecture 3 (Functional Programming) slide 29ff

Because it deals with different syntactic categories

A lot of functions are ultimately functions of expressions which often contains operations (ex : ee (Plus $e1$ $e2$) n)

Because semantic functions are about composition of functions in respect to context.

Last chance for questions