



LLVM Tutorial: Table of Contents

Kaleidoscope: Implementing a Language with LLVM

- 1. Kaleidoscope: Tutorial Introduction and the Lexer
- 2. Kaleidoscope: Implementing a Parser and AST
- 3. Kaleidoscope: Code generation to LLVM IR
- 4. Kaleidoscope: Adding JIT and Optimizer Support
- 5. Kaleidoscope: Extending the Language: Control Flow
- 6. Kaleidoscope: Extending the Language: User-defined Operators
- 7. Kaleidoscope: Extending the Language: Mutable Variables
- 8. Kaleidoscope: Compiling to Object Code
- 9. Kaleidoscope: Adding Debug Information
- 10. Kaleidoscope: Conclusion and other useful LLVM tidbits

Kaleidoscope: Implementing a Language with LLVM in Objective Caml

- 1. Kaleidoscope: Tutorial Introduction and the Lexer
- 2. Kaleidoscope: Implementing a Parser and AST
- 3. Kaleidoscope: Code generation to LLVM IR
- 4. Kaleidoscope: Adding JIT and Optimizer Support
- 5. Kaleidoscope: Extending the Language: Control Flow
- 6. Kaleidoscope: Extending the Language: User-defined Operators
- 7. Kaleidoscope: Extending the Language: Mutable Variables
- 8. Kaleidoscope: Conclusion and other useful LLVM tidbits

Building a JIT in LLVM

- 1. Building a JIT: Starting out with KaleidoscopeJIT
- 2. Building a JIT: Adding Optimizations - An introduction to ORC Layers
- 3. Building a JIT: Per-function Lazy Compilation
- 4. Building a JIT: Extreme Laziness - Using Compile Callbacks to JIT from ASTs
- 5. Building a JIT: Remote-JITing - Process Isolation and Laziness at a Distance

External Tutorials

Tutorial: Creating an LLVM Backend for the Cpu0 Architecture

A step-by-step tutorial for developing an LLVM backend. Under active development at <https://github.com/Jonathan2251/lbd> (please contribute!).

Howto: Implementing LLVM Integrated Assembler

A simple guide for how to implement an LLVM integrated assembler for an architecture.

Advanced Topics

1. [Writing an Optimization for LLVM](#)