Contents

Preface xi

1 Preliminaries 1
1.1 Abstract Syntax Trees 1
1.2 Grammars 3
1.3 Pattern Matching 4
1.4 Recursive Functions 6
1.5 Interpreters 6
1.6 Example Compiler: A Partial Evaluator 9

2 Integers and Variables 13
2.1 The \texttt{L}_{\text{Var}} Language 13
2.2 The \texttt{x86}_\text{toy} Assembly Language 16
2.3 Planning the Trip to \texttt{x86} 22
2.4 Uniquify Variables 25
2.5 Remove Complex Operands 27
2.6 Explicate Control 28
2.7 Select Instructions 30
2.8 Assign Homes 31
2.9 Patch Instructions 32
2.10 Generate Prelude and Conclusion 33
2.11 Challenge: Partial Evaluator for \texttt{L}_{\text{Var}} 33

3 Register Allocation 35
3.1 Registers and Calling Conventions 36
3.2 Liveness Analysis 38
3.3 Build the Interference Graph 42
3.4 Graph Coloring via Sudoku 44
3.5 Patch Instructions 49
3.6 Prelude and Conclusion 50
3.7 Challenge: Move Biasing 52
3.8 Further Reading 55
# Contents

## 4 Booleans and Conditionals

4.1 The $\mathcal{L}_b$ Language

4.2 Type Checking $\mathcal{L}_b$ Programs

4.3 The $\mathcal{C}_b$ Intermediate Language

4.4 The x86 Language

4.5 Shrink the $\mathcal{L}_b$ Language

4.6 Uniquify Variables

4.7 Remove Complex Operands

4.8 Explicate Control

4.9 Select Instructions

4.10 Register Allocation

4.11 Patch Instructions

4.12 Challenge: Optimize Blocks and Remove Jumps

4.13 Further Reading

## 5 Loops and Dataflow Analysis

5.1 The $\mathcal{L}_\text{While}$ Language

5.2 Cyclic Control Flow and Dataflow Analysis

5.3 Mutable Variables and Remove Complex Operands

5.4 Uncover $\text{get!}$

5.5 Remove Complex Operands

5.6 Explicate Control and $\mathcal{C}_\diamond$

5.7 Select Instructions

5.8 Register Allocation

## 6 Tuples and Garbage Collection

6.1 The $\mathcal{L}_\text{Tup}$ Language

6.2 Garbage Collection

6.3 Expose Allocation

6.4 Remove Complex Operands

6.5 Explicate Control and the $\mathcal{C}_{\text{Tup}}$ Language

6.6 Select Instructions and the x86$_\text{Global}$ Language

6.7 Register Allocation

6.8 Prelude and Conclusion

6.9 Challenge: Simple Structures

6.10 Challenge: Arrays

6.11 Uncover $\text{get!}$

6.12 Challenge: Generational Collection

6.13 Further Reading

## 7 Functions

7.1 The $\mathcal{L}_\text{Fun}$ Language

7.2 Functions in x86

7.3 Shrink $\mathcal{L}_\text{Fun}$

7.4 Reveal Functions and the $\mathcal{L}_\text{FunRef}$ Language