

Джереми Сик

ОСНОВЫ КОМПИЛЯЦИИ

инкрементный подход



Санкт-Петербург • Москва • Минск

2024

КРАТКОЕ СОДЕРЖАНИЕ

Предисловие	14
Глава 1. Введение	19
Глава 2. Целые числа и переменные	31
Глава 3. Распределение регистров	55
Глава 4. Булевы выражения и условные конструкции.....	80
Глава 5. Циклы и анализ потоков данных	108
Глава 6. КORTEжи и сборка мусора	123
Глава 7. Функции.....	155
Глава 8. Функции с лексической областью видимости	174
Глава 9. Динамическая типизация	190
Глава 10. Постепенная типизация.....	207
Глава 11. Обобщения.....	228
Приложение.....	241
Список литературы.....	245

ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ.....	14
Благодарности.....	17
ГЛАВА 1. ВВЕДЕНИЕ	19
1.1. Абстрактные синтаксические деревья	20
1.2. Граматики.....	21
1.3. Поиск по шаблону.....	24
1.4. Рекурсивные функции	25
1.5. Интерпретаторы	26
1.6. Пример компилятора: частичный вычислитель	29
ГЛАВА 2. ЦЕЛЫЕ ЧИСЛА И ПЕРЕМЕННЫЕ	31
2.1. Язык $\mathcal{L}_{Var}$	31
2.1.1. Расширение интерпретаторов через переопределение методов.....	33
2.1.2. Определительный интерпретатор для $\mathcal{L}_{Var}$	34
2.2. Язык ассемблера $x86_{Int}$	37
2.3. Планирование перехода на уровень $x86$	41
2.3.1. Промежуточный язык $\mathcal{C}_{Var}$	44

2.4. Создание уникальных имен переменных	45
2.5. Удаление составных операндов.....	46
2.6. Явное выражение порядка выполнения.....	48
2.7. Выбор инструкций.....	50
2.8. Размещение переменных.....	52
2.9. Проход <code>patch_instructions</code>	53
2.10. Генерирование пролога и эпилога.....	53
2.11. Упражнения повышенной сложности: частичный вычислитель для $\mathcal{L}_{Var}$	54
ГЛАВА 3. РАСПРЕДЕЛЕНИЕ РЕГИСТРОВ	55
3.1. Регистры и соглашения о вызовах	56
3.2. Анализ живых переменных	60
3.3. Построение графа перекрытий.....	63
3.4. Раскраска графа и sudoku.....	65
3.5. Проход <code>patch_instructions</code>	72
3.6. Пролог и эпилог.....	72
3.7. Упражнение повышенной сложности: смещение команд <code>movq</code>	75
3.8. Дополнительная литература	78
ГЛАВА 4. БУЛЕВЫ ВЫРАЖЕНИЯ И УСЛОВНЫЕ КОНСТРУКЦИИ	80
4.1. Язык $\mathcal{L}_{If}$	81
4.2. Программы проверки типов $\mathcal{L}_{If}$	84
4.3. Промежуточный язык $\mathcal{C}_{If}$	87
4.4. Язык <code>x86_{If}</code>	88
4.5. Сжатие языка $\mathcal{L}_{If}$	91
4.6. Проход <code>uniquify_exp</code>	91

4.7. Проход <code>remove_complex_operands</code>	92
4.8. Проход <code>explicate_control</code>	92
4.8.1. Функции <code>explicate_tail</code> и <code>explicate_assign</code>	95
4.8.2. Создание блоков	96
4.8.3. Явное выражение предиката	96
4.8.4. Взаимодействия между <code>explicate_control</code> и <code>shrink</code>	98
4.9. Проход <code>select_instructions</code>	99
4.10. Распределение регистров	100
4.10.1. Анализ живых переменных	100
4.10.2. Построение графа перекрытий	102
4.11. Проход <code>patch_instructions</code>	102
4.12. Упражнение повышенной сложности: оптимизация блоков и удаление переходов	104
4.12.1. Оптимизация блоков	104
4.12.2. Удаление переходов	106
4.13. Дополнительная литература	107
ГЛАВА 5. ЦИКЛЫ И АНАЛИЗ ПОТОКОВ ДАННЫХ	108
5.1. Язык $\mathcal{L}_{while}$	109
5.2. Циклическое выполнение и анализ потоков данных	112
5.3. Изменяемые переменные и удаление составных операндов	116
5.4. Открытие <code>get!</code>	118
5.5. Удаление составных операндов	119
5.6. Проход <code>explicate_control</code> и $\mathcal{C}_\circ$	120
5.7. Проход <code>select_instructions</code>	121
5.8. Распределение регистров	121

ГЛАВА 6. КОРТЕЖИ И СБОРКА МУСОРА	123
6.1. Язык $\mathcal{L}_{Typ}$	123
6.2. Сборка мусора	128
6.2.1. Копирующий сборщик мусора с двумя пространствами	128
6.2.2. Копирование графа по алгоритму Чени	130
6.2.3. Представление данных	132
6.2.4. Реализация сборщика мусора	134
6.3. Проход <code>expose_allocation</code>	135
6.4. Проход <code>remove_complex_operands</code>	136
6.5. Проход <code>explicate_control</code> и язык $\mathcal{C}_{Typ}$	137
6.6. Проход <code>select_instructions</code> и язык $x86_{Global}$	138
6.7. Распределение регистров	142
6.8. Пролог и эпилог	142
6.9. Упражнение повышенной сложности: простые структуры	144
6.10. Упражнение повышенной сложности: массивы	146
6.10.1. Представление данных	149
6.10.2. Разрешение перегрузки	150
6.10.3. Проверка границ	150
6.10.4. Выделение памяти	151
6.11. Обработка <code>get!</code>	151
6.11.1. Удаление составных операндов	151
6.11.2. Явное выражение управляющих операций	151
6.11.3. Выбор инструкций	151
6.12. Упражнение повышенной сложности: сборка мусора с поколениями	152
6.13. Дополнительная литература	153

ГЛАВА 7. ФУНКЦИИ	155
7.1. Язык $\mathcal{L}_{Fun}$	155
7.2. Функции в x86.....	160
7.2.1. Соглашения о вызовах	160
7.2.2. Эффективные хвостовые вызовы.....	162
7.3. Проход shrink для $\mathcal{L}_{Fun}$	164
7.4. Проход reveal_functions и язык $\mathcal{L}_{FunRef}$	164
7.5. Проход limit_functions	164
7.6. Проход remove_complex_operands	165
7.7. Проход explicate_control и язык $\mathcal{C}_{Fun}$	166
7.8. Проход select_instructions и язык $x86_{callq}^{Def}$	167
7.9. Распределение регистров	170
7.9.1. Анализ живых переменных	170
7.9.2. Построение графов перекрытий	170
7.9.3. Проход allocate_registers	170
7.10. Проход patch_instructions.....	171
7.11. Пролог и эпилог	171
7.12. Пример преобразования.....	173
ГЛАВА 8. ФУНКЦИИ С ЛЕКСИЧЕСКОЙ ОБЛАСТЬЮ ВИДИМОСТИ	174
8.1. Язык $\mathcal{L}_\lambda$	176
8.2. Присваивание и функции с лексической областью видимости.....	179
8.3. Проход convert_assignments.....	180
8.4. Проход convert_to_closures	182
8.5. Пример преобразования	184
8.6. Выделение памяти	185

8.7. Проход <code>explicate_control</code> и $\mathcal{C}_{Clos}$	185
8.8. Проход <code>select_instructions</code>	186
8.9. Упражнение повышенной сложности: оптимизация замыканий	187
8.10. Дополнительная литература	189
ГЛАВА 9. ДИНАМИЧЕСКАЯ ТИПИЗАЦИЯ	190
9.1. Язык $\mathcal{L}_{Dyn}$	191
9.2. Представление тегированных значений	195
9.3. Язык $\mathcal{L}_{Any}$	195
9.4. Проход <code>cast_insert</code> : компиляция $\mathcal{L}_{Dyn}$ в $\mathcal{L}_{Any}$	200
9.5. Проход <code>reveal_casts</code>	201
9.6. Удаление сложных операндов	202
9.7. Проход <code>explicate_control</code> и $\mathcal{C}_{Any}$	202
9.8. Проход <code>select_instructions</code>	203
9.9. Распределение регистров для $\mathcal{L}_{Any}$	205
ГЛАВА 10. ПОСТЕПЕННАЯ ТИПИЗАЦИЯ	207
10.1. Проверка типов $\mathcal{L}_?$	209
10.2. Интерпретация $\mathcal{L}_{Cast}$	215
10.3. Проход <code>cast_insert</code>	219
10.4. Проход <code>lower_casts</code>	220
10.5. Кортежи и их посредники	222
10.6. Проход <code>reveal_casts</code>	224
10.7. Преобразование замыканий	224
10.8. Проход <code>select_instructions</code>	224
10.9. Дополнительная литература	227

ГЛАВА 11. ОБОБЩЕНИЯ.....	228
11.1. Компиляция обобщений	235
11.2. Обработка инстанцирования	236
11.3. Стирание обобщенных типов	238
ПРИЛОЖЕНИЕ	241
A.1. Интерпретаторы	241
A.2. Вспомогательные функции.....	241
A.3. Краткий список инструкций x86	243
СПИСОК ЛИТЕРАТУРЫ.....	245