

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Костюк Ю. Л.

**ТЕОРИЯ АВТОМАТОВ И
ФОРМАЛЬНЫХ ЯЗЫКОВ**

Лекция 6

ГЕНЕРАЦИЯ КОМАНД В КОМПИЛЯТОРЕ

Распределение памяти при работе программы:

- 1) программа из машинных команд;
- 2) константы, используемые в машинных командах;
- 3) глобальные статические переменные;
- 4) магазин для вызовов процедур и динамические переменные.

Операционная система (ОС) считывает с диска файл (исполняемый модуль - программа и константы) и передает программе управление.

Области памяти для переменных выделяются по запросу при выполнении программы.

Последняя команда программы - возврат управления в ОС.

Одноадресная система команд

Формат команд:

<код операции> <операнд>

<операнд> - адрес в памяти (относительно начала выделенной области).

Регистр сумматора (сумматор) содержит второй операнд, результат операции – в сумматоре.

Набор команд:

Add <операнд> – сложение,

Sub <операнд> – вычитание,

Mul <операнд> – умножение,

Div <операнд> – деление,

Load <операнд> – запись из операнда в сумматор,

St <операнд> – запись из сумматора в операнд.

Генерация команд по сгенерированной ОПС интерпретатором

В интерпретаторе используется магазин, в него записываются операнды (номера ячеек памяти) и результаты вычислений.

Если результат выполнения команды находится в сумматоре, то в магазин записывается нуль.

Переменная k напрямую ссылается на тот элемент магазина, который содержит нуль.

ОПС просматривается слева направо.

Если очередной элемент в ОПС – операнд, то он записывается в магазин.

Если очередной элемент в ОПС – операция, то из магазина считываются операнды и генерируются команды, выполняющие операцию.

Два верхних операнда в магазине - a и b .

Генерация команд для арифметических выражений

1. Если $k = 0$, то генерируются команды:

Load a

Op b

где Op – команда Add, Sub, Mul или Div.

В магазин записывается нуль, а в переменную k – ссылка на верхний элемент магазина.

2. Если $k \neq 0$, $a = 0$, то генерируется команда:

Op b

где Op – команда Add, Sub, Mul или Div.

В магазин записывается нуль, а в переменную k – ссылка на верхний элемент магазина.

3. Если $k \neq 0$, $b = 0$, то генерируются команды:

Op b

где Op – команда Add или Mul, или команды:

St t

Load a

Op t

где Op – команда Sub или Div,

t – дополнительная временная переменная. В магазин записывается 0, k – ссылка на верхний элемент магазина.

4. Если $k \neq 0$, $a \neq 0$, $b \neq 0$, то генерируются команды:

St t

Load a

Op b

где Op – команда Add, Sub, Mul или Div, t – дополнительная временная переменная. В магазин записывается 0, в элемент магазина с номером k записывается t, k – ссылка на верхний элемент магазина.

5. Для операции :=, если $k = 0$, то генерируются команды:

Load b

St a

6. Для операции :=, если $k \neq 0$, $b \neq 0$, то генерируются команды:

St t

Load b

St a

где t – дополнительная временная переменная. При этом в элемент магазина с номером k записывается t .

7. Для операции :=, если $k \neq 0$, $b = 0$, то генерируется команда:

St a

$k := 0$, в магазин ничего не записывается.

Пример генерации команд

Пусть задана ОПС: $x \ a \ b - c \ d \ e / + * :=$

Получена из оператора: $x := (a - b) * (c + d / e)$

Шаги работы генератора команд:

1. Операция $-$, $k = 0$, в магазине: $x \ a \ b$

генерируемые команды:

Load a

Sub b

$k := 2$

2. Операция $/$, $k = 2$, в магазине: $x \ 0 \ c \ d \ e$

генерируемые команды:

St t

Load d

Div e

2-я ячейка магазина $:= t$, $k := 4$

3. Операция $+$, $k = 4$, в магазине: $x \ t \ c \ 0$

генерируемая команда:

Add c

$k := 3$

4. Операция $*$, $k = 3$, в магазине: $x \ t \ 0$

генерируемая команда:

Mul t

$k := 2$

5. Операция $:=$, $k = 2$, в магазине: $x \ 0$

генерируемая команда:

St x

$k := 0$, магазин пуст.

Результат:

Load a

Sub b

St t

Load d

Div e

Add c

Mul t

St x

Выделение памяти для массивов

Команда *прерывания* передает управление в область памяти с операционной системой:

Itr <номер прерывания>

Номер прерывания задает конкретное действие операционной системе, например, выделение памяти.

При этом в *регистре сумматора* должен быть задан размер области памяти.

При выполнении команды **Itr** выполняется передача управления в операционную систему и запоминается адрес команды, расположенной следом за командой **Itr**, в специальном *регистре адреса*.

После этого выполняется программа операционной системы, она выделяет область памяти, адрес ее начала записывает в *регистр сумматора* и возвращает управление по адресу, записанному в *регистре адреса*.

Генерация команд выделения памяти

Обратная польская строка с выделением памяти для одномерного массива:

M n <m1>

M – ссылка на паспорт массива,

n – количество элементов массива,

<m1> - операция выделения памяти.

В паспорте массива:

– адрес нулевого элемента массива;

– длина одного элемента массива;

– количество элементов массива

записываются при выполнении операции <m1>.

При интерпретации операции **<m1>** генерируются команды выделения памяти и формирования паспорта массива:

Load n

St M + 2v {количество элементов n}

Mul <длина элемента> {размер памяти в байтах}

Itr <выделение памяти> {прерывание}

St M {ссылка на начало массива}

Load <длина элемента>

St M + v {длина элемента массива}

Здесь **v** – длина в байтах целочисленного значения.

Расширение модели процессора. Вид команд:

<код операции> <признак> <операнд>

Если признак равен 0, то команда выполняется обычным образом, а если 1, то используется **косвенная адресация**, когда операнд в команде – это адрес в памяти, где находится адрес, ссылающийся на другую ячейку памяти, содержащую значение операнда.

В магазине интерпретатора для каждой ячейки магазина дополнительно записывался признак (0 или 1). Тогда, при генерации любой команды с операндом из магазина, в команду записывается операнд и соответствующий признак.

Генерация команд, реализующих операцию индексирования

Два верхних операнда в магазине:

a - ссылка на паспорт массива,

b – индекс элемента массива.

В переменной k находится ссылка на элемент магазина, содержащий 0.

$t1, t2$ – дополнительные временные переменные.

Если $b = 0$, то генерируются команды:

Mul 0 $a + v$

Add 0 a

St 0 $t1$

В магазин записывается $t1$ с признаком 1, а также $k := 0$.

Если $b \neq 0$, $k = 0$, то генерируются команды:

Load 0 b

Mul 0 $a + v$

Add 0 a

St 0 t1

В магазин записывается t1 с признаком 1, а также $k := 0$.

Если $b \neq 0$, $k \neq 0$, то генерируются команды:

St 0 t1

Load 0 b

Mul 0 $a + v$

Add 0 a

St 0 t2

В элемент магазина по ссылке k записывается t1, в верхний элемент магазина записывается t2 и признак для него 1, а также $k := 0$.

Пример генерации команд

Пусть задана ОПС: $M[j + a] + \langle i \rangle L[i - d] - \langle i \rangle :=$

- оператор: $M[j + a] := L[i - d]$

Шаги работы генератора команд:

Операция $+$, $k = 0$, в магазине: $M(0) \ j(0) \ a(0)$

(в скобках указаны признаки элементов магазина)

генерируемые команды:

Load $0 \ j$

Add $0 \ a$

$k := 2$

Операция $\langle i \rangle$, $k = 2$, в магазине: $M(0) \ 0(0)$

генерируемые команды:

Mul $0 \ M + v$

Add $0 \ M$

St $0 \ t1$

$k := 0$

Операция $-$, $k = 0$, в магазине: $t1(1) \quad L(0) \quad i(0) \quad d(0)$

генерируемые команды:

Load $0 \quad i$

Sub $0 \quad d$

$k := 3$

Операция $<i>$, $k = 3$, в магазине: $t1(1) \quad L(0) \quad 0(0)$

генерируемые команды:

Mul $0 \quad L + v$

Add $0 \quad L$

St $0 \quad t2$

$k := 0$

Операция $:=$, $k = 3$, в магазине: $t1(1) \quad t2(1)$

генерируемые команды:

Load $1 \quad t2$

St $1 \quad t1$

$k := 0$ магазин пуст.

Результат:

Load 0 j

Add 0 a

Mul 0 $M + v$

Add 0 M

St 0 t1

Load 0 i

Sub 0 d

Mul 0 $L + v$

Add 0 L

St 0 t2

Load 1 t2

St 1 t1