Генерация внутренних представлений

Гладков Артемий Николаевич

06.11.2024

Генерация внутренних представлений

После синтаксического анализа необходимо получить некоторое внутренне представление программы для дальнейшей трансляции в объектный код или интерпретации.
Этот код универсален для всех машин, на которых происходит исполнение.

Способы внутреннего представления программы

- Многоадресный код с явно именуемыми результатами.
- Многоадресный код с неявно именуемыми результатами.
- Связанные списочные структуры, представляющие синтаксические деревья
- Префиксная запись
- Постфиксная запись (польская инверсная запись)

Способы внутреннего представления программы

- Многоадресный код с явно именуемыми результатами.
- Многоадресный код с неявно именуемыми результатами.
- Связанные списочные структуры, представляющие синтаксические деревья
- Префиксная запись
- Постфиксная запись (польская инверсная запись), сокращённо **ПОЛИЗ**

ПОЛИЗ для выражений

В инфиксной записи:

$$a * (b + c) - (d - e)/f$$

в постфиксной:

$$abc + *de - f/-$$

Порядок операндов сохранился, а для операций учтён приоритет, и исчезли скобки.
Выражения, записанные в ПОЛИЗ-е удобно вычислять с помощью стека.

Генерация ПОЛИЗ-а по выражению:

- Если E — операнд, то T — его ПОЛИЗ;
- ПОЛИЗом $E_1 \theta E_2$, где θ — бинарная операция, а E_1, E_2 — выражения, является запись $E_1' E_2' \theta$, где E_1' и E_2' — ПОЛИЗ выражений E_1 и E_2;
- ПОЛИЗом выражения θE, где θ — знак унарной операции, а E — операнд, является запись $E' \theta$, где E' — ПОЛИЗ выражения E;
- ПОЛИЗом выражения (E) является ПОЛИЗ выражения E.

Вычисление выражения в ПОЛИЗЕ

Выражение прочитывается один раз слева направо и:

- если очередной элемент ПОЛИЗа — это операнд, то его значение заносится в стек;
- если очередной элемент ПОЛИЗа — это операция, то на верхушке стека сейчас находятся ее операнды; они извлекаются из стека, над ними выполняется операция, результат снова заносится в стек;
- когда выражение, записанное в ПОЛИЗе, прочитано, в стеке останется один элемент – это значение всего выражения.

Операторы языка в ПОЛИЗе

Присваивание

$$I := E$$

будет записано в ПОЛИЗе как

$$\tilde{I} E :=$$

где $:=$ — это двухместная операция, а $\tilde{I}$ и E — операнды. $\tilde{I}$ — означает, что операндом является адрес I, а не её значение.

Операторы языка в ПОЛИЗе

Оператор перехода означает, что процесс интерпретации надо продолжить с того элемента ПОЛИЗа, который указан как операнд операции перехода.

Чтобы можно было ссылаться на элементы ПОЛИЗа, будем считать, что все они перенумерованы, начиная с 1 (допустим, занесены в массив). Пусть ПОЛИЗ оператора, помеченного меткой L, начинается с номера p, тогда оператор перехода `goto L` в ПОЛИЗе можно записать как

$$p!$$

где ! — операция выбора элемента ПОЛИЗа, номер которого равен p.

Операторы языка в ПОЛИЗе

Оператор перехода означает, что процесс интерпретации надо продолжить с того элемента ПОЛИЗа, который указан как операнд операции перехода.
Чтобы можно было ссылаться на элементы ПОЛИЗа, будем считать, что все они перенумерованы, начиная с 1 (допустим, занесены в массив).
Пусть ПОЛИЗ оператора, помеченного меткой L, начинается с номера p, тогда оператор перехода `goto L` в ПОЛИЗе можно записать как

$$p!$$

где ! — операция выбора элемента ПОЛИЗа, номер которого равен p.

Операторы языка в ПОЛИЗе

Оператор перехода означает, что процесс интерпретации надо продолжить с того элемента ПОЛИЗа, который указан как операнд операции перехода.
Чтобы можно было ссылаться на элементы ПОЛИЗа, будем считать, что все они перенумерованы, начиная с 1 (допустим, занесены в массив).
Пусть ПОЛИЗ оператора, помеченного меткой L, начинается с номера p, тогда оператор перехода `goto L` в ПОЛИЗе можно записать как

$$p!$$

где ! — операция выбора элемента ПОЛИЗа, номер которого равен p.

Операторы языка в ПОЛИЗе

Условный оператор перехода

Введем вспомогательную операцию – условный переход «по лжи» с семантикой:

$$if\,(not\,B)\;then\;goto\;L$$

Это двухместная операция с операндами B и L. Обозначим ее $!F$, тогда в ПОЛИЗе она будет записана как:

$$B\;p\;!F$$

аналогично p – номер элемента, с которого начинается ПОЛИЗ оператора, помеченного меткой L.

Операторы языка в ПОЛИЗе

Тогда **условный оператор**

$$\textbf{if } B \textbf{ then } S1 \textbf{ else } S2$$

будет записан, как

$$\textbf{if}(\textbf{not } B)\textbf{then } goto\ L2; S1; goto\ L3; \textbf{L2: } S2; \textbf{L3: } \ldots,$$

а в ПОЛИЗе:

$$B\ p2\ !F\ S1\ p3\ !\ \underset{\uparrow}{S2}\ \ldots\ \underset{\uparrow}{}$$
$$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad p2 \quad p3$$

Операторы языка в ПОЛИЗе

Тогда **оператор цикла**

$$\textbf{while } B \textbf{ do } S$$

будет записан, как

$$\textbf{if}(\textbf{not } B)\textbf{then } goto\ L2; S1; goto\ L3; \textbf{L2: } S2; \textbf{L3: } \ldots,$$

а в ПОЛИЗе:

$$\underset{\uparrow}{B}\ p1\ !F\ S\ \underset{\uparrow}{p0}\ !\ldots$$
$$p0 \qquad\qquad p1$$

Операторы языка в ПОЛИЗе

И, наконец, оператор ввода **read** (I) в ПОЛИЗе будет записан как $I\ R$; оператор вывода **write** (E) – как $E\ W$.

Генерация ПОЛИЗа

Генерацию ПОЛИЗа удобно совместить с синтаксическим анализатором.

Операторы языка в ПОЛИЗе

Рассмотрим грамматику:

$E \to T\{+T\}$

$T \to F\{*F\}$

$F \to a \mid b \mid (E)$

Тогда грамматика с действиями по переводу этого выражения в ПОЛИЗ будет такой:

$E \to T\{+T < put(" + ") >\}$

$T \to F\{*F < put(" * ") >\}$

$F \to a < put("a") > \mid b < put("b") > \mid (E)$

Результат генерации ПОЛИЗа

```
...
while a < b do begin
  {if beleberda just commentary
  write(a);}
  a := a + 1;
  c := c * a
end  &
...
```

Преобразовав, получим:

$$\dots \underset{\substack{\uparrow \\ 11}}{a}\; b < 26\, !F\, \tilde{a}\, a\, 1 + := \tilde{c}\, c\, a * := 11\, !\, \underset{\substack{\uparrow \\ 26}}{\dots}$$

Что дальше?

- Посмотреть видео с моей реализацией Генерации внутренних представлений: <ссылка>.

Что дальше?

- Посмотреть видео с моей реализацией Генерации внутренних представлений: <ссылка>.
- Доработать текущее решение лабораторной работы №3. Добавить в него генерацию ПОЛИЗа.

Будущие лекции

- Занятие 5: первая половина пары — исполнение кода.
- Занятие 6 — приём Лабораторных. Или конец материала, если что-то не успеем.