

Vorlesung “Compilerbau” WS 2011/2012

11. Übungsblatt

Abgabe: 25. Januar 2012 bis 10.45 Uhr

Aufgabe 1 (2 Punkte)

Übersetzen Sie den arithmetischen Ausdruck $\mathbf{a} * (\mathbf{b} + \mathbf{c})$ in Drei-Adress-Code.

Aufgabe 2 (8 Punkte)

Entwickeln Sie für die folgende Teilgrammatik ein Übersetzungsschema zur Erzeugung von Drei-Adress-Code.

$$\begin{aligned} S &\rightarrow \mathbf{id} := E \\ E &\rightarrow E_1 + E_2 \\ E &\rightarrow E_1 * E_2 \\ E &\rightarrow - E_1 \\ E &\rightarrow (E_1) \\ E &\rightarrow \mathbf{id} \end{aligned}$$

Verwenden Sie die Funktion *newtemp()* zur Generierung temporärer Variablen.

Aufgabe 3 (4 Punkte)

Erweitern Sie das Übersetzungsschema für die Grammatik aus Aufgabe 37 so, dass die folgenden Zusatzbedingungen berücksichtigt werden. Wir nehmen nun an, dass die Ausdrücke vom Typ **integer** und **real** sein können. Mittels der Funktion *inttoreal()* sollen bei Bedarf Zahlen vom Typ **integer** in Zahlen des Typs **real** umgewandelt werden. Ferner sollen auch anstelle der Operatoren $+$ und $*$ die passenden Operatoren **int+**, **real+**, **int*** und **real*** verwendet werden.

Aufgabe 4 (Zusatzaufgabe) (16 Punkte)

Für ein eindimensionales Array A mit Elementen der Länge w beginnt das i -te Element am Speicherplatz

$$base + (i - low) \cdot w$$

wobei low die untere Grenze der Array-Indizes und $base$ die relative Adresse des Speicherplatzes ist, die für das Array zugewiesen wurde. Durch Umformulierung erhalten wir

$$i \cdot w + (base - low \cdot w)$$

Der Teilausdruck $c = base - low \cdot w$ kann zur Übersetzungszeit ausgewertet werden, wenn die Deklaration des Arrays erkannt wird. Wir nehmen an, dass c in dem Symboltabelleneintrag für A gespeichert wird, so dass die relative Adresse von $A[i]$ durch einfaches Addieren von $i \cdot w$ zu c berechnet werden kann.

Solche Vorberechnungen zur Übersetzungszeit können auch für mehrdimensionale Arrays durchgeführt werden. Für ein zweidimensionales Array, welches in zeilenorientierter Form gespeichert wird, kann die relative Adresse von $A[i_1, i_2]$ mit der Formel

$$base + ((i_1 - low_1) \cdot n_2 + i_2 - low_2) \cdot w$$

berechnet werden, wobei low_1 und low_2 die unteren Grenzen der Werte i_1 und i_2 sind und n_2 die Anzahl der Werte ist, die i_2 annehmen kann (d.h., falls $high_2$ die obere Grenze der Werte von i_2 ist, dann ist $n_2 = high_2 - low_2 + 1$). Mit der Annahme, dass i_1 und i_2 die einzigen Werte sind, die zur Übersetzungszeit nicht bekannt sind, können wir den obigen Ausdruck umformulieren zu

$$((i_1 \cdot n_2) + i_2) \cdot w + (base - ((low_1 \cdot n_2) + low_2) \cdot w)$$

Der letzte Term $c = (base - ((low_1 \cdot n_2) + low_2) \cdot w)$ in diesem Ausdruck kann zur Übersetzungszeit bestimmt werden.

Verallgemeinern Sie die obigen Ausdrücke und Umformulierungen auf k -dimensionale Arrays, welche in zeilenorientierter Form gespeichert werden. Entwickeln Sie für die folgende Grammatik ein Übersetzungsschema, mit dem u.a. ein beliebiger Arrayausdruck $A[i_1, \dots, i_k]$ in den 3-Adress-Code $A[offset]$ übersetzt wird, wobei $offset$ den korrekten Abstand zur ersten Speicherzelle gemäß der Indizierung i_1, \dots, i_k darstellt.

$$\begin{array}{ll}
 S & \rightarrow L := E \\
 E & \rightarrow E_1 + E_2 \\
 E & \rightarrow (E_1) \\
 E & \rightarrow L \\
 L & \rightarrow Elist] \\
 L & \rightarrow \mathbf{id} \\
 Elist & \rightarrow Elist_1 , E \\
 Elist & \rightarrow \mathbf{id} [E
 \end{array}$$

Verwenden Sie die folgenden Attribute und Hilfsfunktionen sowie beliebige weitere Attribute:

- das Attribut $L.place$ für den zugehörigen Namen bzw. Speicherzelle,
- das Attribut $L.offset$ zum Speichern des Offsets. Für einfache Namen kann man als Indikator $L.offset = -1$ setzen,
- das Attribut $Elist.ndim$ für die Dimension des Arrays,
- das Attribut $Elist.array$ für den Namen bzw. die erste Speicherzelle des Arrays,
- die Funktion $limit(array, j)$ liefert n_j , die Anzahl der Elemente entlang der j -ten Dimension des Arrays für $array$,
- die Funktion $width(array)$ liefert die Länge w eines Array-Elements,
- die Funktion $c(array)$ liefert den zur Übersetzungszeit bestimmbaren Wert des Terms c (siehe oben).