GRAMATIKA IN PREVAJANJE ARITMETIČNIH IZRAZOV JEZIKA PL/1

V. Batagelj
Institut "Jožef Stefan", Ljubljana
Jugoslavija

POVZETEK : V sestavku je podan del LL(1) gramatike za jezik PL/1 , ki opisuje aritmetične izraze.

ABSTRACT : In the paper is given a LL(1) grammar for arithmetic expressions of PL/1 .

Gramatika je del LL(1) gramatike za PL/1 . Pri pisanju gramatike za aritmetične izraze smo se oprli na sestavek (A). V naslednjem spisku produkcij vstopajo v gramatiko PL/1 produkcije P3 - P20 . Prve tri produkcije so dodane zato, da primer ne bi visel preveč v zraku:

P0	⟨ar. stavek⟩	::=	IME ⟨imenad⟩ ⟨imenek⟩
P1		::=	NAPAKA
P2	⟨imenek⟩	::=	: ⟨ar. stavek⟩
P3	⟨predstavek⟩	::=	, ⟨spremenljivka⟩ ⟨predstavek⟩
P4		::=	= ⟨izraz.1⟩
P5		::=	NAPAKA
P6	⟨spremenljivka⟩	::=	IME ⟨imenad⟩
P7		::=	NAPAKA
P8	⟨imenad⟩	::=	. ⟨spremenljivka⟩
P9	⟨lista⟩	::=	(⟨izraz.1⟩ ⟨nadime⟩
P10		::=	ε
P11	⟨nadime⟩	::=	, ⟨izraz.1⟩ ⟨nadime⟩
P12		::=	)
P13		::=	NAPAKA
P14	⟨izraz.i⟩	::=	KONSTANTA ⟨ostanek.i.8⟩
P15		::=	IME ⟨imenad⟩ ⟨ostanek.i.8⟩
P16		::=	OPERATOR.7 ⟨izraz.8⟩ ⟨ostanek.i.7⟩ $i \leq 7$

P17 ::= (<izraz.1>) <ostanek.i.8>

P18 ::= NAPAKA

P19 <ostanek.i.j> ::= OPERATOR.m <izraz.m+1> <ostanek.i.m> $i \leq m \leq j \wedge m \neq 7$

P20 ::= ε

pri tem je $1 \leq i \leq 9$, $1 \leq j \leq 8$; m pa je takoimenovano prioritetno število omejeno z i in j .

Operatorji so takole določeni:

OPERATOR.1 - .OR.

OPERATOR.2 - .AND.

OPERATOR.3 - osnovne PL/1 relacije

OPERATOR.4 - stikanje (konkatenacija)

OPERATOR.5 - + in -

OPERATOR.6 - * in /

OPERATOR.7 - .NOT. , prefiksni + in prefiksni -

OPERATOR.8 - **

Gornja gramatika se loči od običajne LL(1) gramatike v dveh stvareh.

1. v isto skupino produkcij je možnih več vhodov - na primer v P3 - P5 lahko pridemo po neterminalu <imenek> ali pa po neterminalu <predstavek> .
2. nekateri neterminali so indeksirani - produkcije P14 - P20 . Vsaka od teh produkcij določa pravzaprav nek razred produkcij. V običajni gramatiki bi namreč stala za vsako dovoljeno vrednost indeksov i in j posebna produkcija.

Oba prijema omogočita občutno zmanjšanje števila produkcij in s tem tudi prostora, ki ga zavzemajo. Zato pa potrebujemo poseben aparat za delo z indeksi.

Skupino produkcij končuje produkcija oblike

<neterminal> ::= ε

ali

<neterminal> ::= NAPAKA

V prvem primeru razpoznavalnik nadaljuje delo brez pomika vhoda; v drugem pa sproži ustrezno rutino.

Med prevajanjem generiramo izraze v vmesnem jeziku. Vpeljemo naslednje oznake

(#,A) - ime A "kandidira" za ime polja, rutine ali strukture

(.,A) - ime A je ime strukture

(O,A) - A je običajno ime katerega naslovov lahko določimo v času prevajanja

(OP,A,B)- operacija OP ima prvi argument A in drugi argument B

potem lahko takole prikažemo prevajanje aritmetičnega izraza:

A(I),B.C = (X + Y * W) + U * V / 2 ** N ;

VRH SKLADA	VHOD	PRODUKCIJA	GLOBINA SEMANTIČNEGA SKLADA	VSEBINA SKLADA	ZAP. ŠT. TROJČKA	MATRIKA TROJČKOV
⟨ar. stavek⟩	A	P0	1	(#,A)		
⟨imenad⟩	(	P9				
⟨izraz.1⟩	I	P15	2	(#,I)		
⟨imenad⟩	)	P10	2	(0,I)		
⟨ostanek.18⟩	)	P20				
⟨nadime⟩	)	P12	1	(0,M1)	1	(#,A,I)
⟨imenek⟩	,	P3	1	(=,M1,)		
⟨spremenljivka⟩	B	P6	2	(#,B)		
⟨imenad⟩	.	P8	2	(.,B)		
⟨spremenljivka⟩	C	P6	3	(#,C)		
⟨imenad⟩	=	P10	2	(0,B.C)		
⟨predstavek⟩	=	P4	2	(=,B.C,)		
⟨izraz.1⟩	(	P17				
⟨izraz.1⟩	X	P15	3	(#,X)		
⟨imenad⟩	+	P10	3	(0,X)		
⟨ostanek.18⟩	+	P19	3	(+,X,)		
⟨izraz.6⟩	Y	P15	4	(#,Y)		
⟨imenad⟩	*	P10	4	(0,Y)		
⟨ostanek.68⟩	*	P19	4	(*,Y,)		
⟨izraz.7⟩	W	P15	5	(#,W)		
⟨imenad⟩	)	P10	5	(0,W)		
⟨ostanek.78⟩	)	P20	4	(0,M2)	2	(*,Y,W)
⟨ostanek.66⟩	)	P20	3	(0,M3)	3	(+,X,M2)
⟨ostanek.15⟩	)	P20				
)	)	TEST				
⟨ostanek.18⟩	+	P19	3	(+,M3,)		
⟨izraz.6⟩	U	P15	4	(#,U)		
⟨imenad⟩	*	P10	4	(0,U)		
⟨ostanek⟩	*	P19	4	(*,U,)		
⟨izraz.7⟩	V	P15	5	(#,V)		
⟨imenad⟩	/	P10	5	(0,V)		
⟨ostanek.78⟩	/	P20	4	(0,M4)	4	(*,U,V)
⟨ostanek.66⟩	/	P19	4	(/,M4)		
⟨izraz.7⟩	2	P14	5	(0,2)		
⟨ostanek.78⟩	**	P19	5	(**,2,)		
⟨izraz.9⟩	N	P15	6	(#,N)		

VRH SKLADA	VHOD	PRODUKCIJA	GLOBINA SEMANTIČNEGA SKLADA	VSEBINA SEMANTIČNEGA SKLADA	ZAP. ŠT. TROJČKA	MATRIKA TROJČKOV
⟨imenad⟩	;	P10	6	(O,N)		
⟨ostanek.98⟩	;	P20	5	(O,M5)	5	(**,2,N)
⟨ostanek.78⟩	;	P20	4	(O,M6)	6	(/,M4,M5)
⟨ostanek.66⟩	;	P20	3	(O,M7)	7	(+,M3,M6)
⟨ostanek.15⟩	;	P20	2	(O,M8)	8	(=,B.C,M7)
			1	(O,M9)	9	(=,M1,M8)

V matriki M dobimo prevod izraza v vmesnem jeziku. Nad to matriko ponavadi izvedemo še optimizacijo in končno generiramo kod.

Iz gornje tabele je tudi razvidno, kaj mora v grobem storiti posamezni produkciji ustrezna semantična rutina.

LITERATURA

(A) P.M. LEWIS II, D.J. ROSENKRANTZ : An Algol Compiler Designed Using Automata Theory, Symposium on Computers and Automata, Brooklyn, 1971

(B) DAVID GRIES : Compiler Construction for Digital Computers, JW, New York 1971