

Fundamentele Limbajelor de Programare

Gramatici de operatori cu precedență

Traian-Florin Șerbănuță

UNIBUC

19 martie 2021

Gramatici de operatori cu precedențe

Definiție

O gramatică independentă de context se numește gramatică de operatori dacă:

- ▶ Nu are producții vide $A ::=$
- ▶ Nu are terminale adiacente în partea dreaptă $A ::= BC$

Exemplu rău

$$E ::= E \ A \ E \mid -E \mid (E) \mid x$$
$$A ::= + \mid - \mid * \mid / \mid ^$$

Exemplu bun

$$E ::= E + E \mid E - E \mid E * E \mid E / E \mid E ^ E \mid - E \mid T$$
$$T ::= (E) \mid \text{id} \mid \text{nat}$$

Adăugăm precedente și attribute de asociativitate

```
E ::= T  
    > E ^ E (right)  
    > - E  
    > E * E (left) | E / E (left)  
    > E + E (left) | E - E (left)
```

Calculăm tabela de precedențe

$$E ::= T$$
$$> E \wedge E \text{ (right)}$$

> - E

$$> E * E \text{ (left)} \mid E / E \text{ (left)}$$
$$> E + E \text{ (left)} \mid E - E \text{ (left)}$$
[illegible]

Adăugăm precedentele în șirul de analizat

	T	^	0-	*	/	+	-	\$
T		>		>	>	>	>	>
^	<	<	>	>	>	>	>	>
0-	<	<	<	>	>	>	>	>
	<	<	<	>	>	>	>	>
/	<	<	<	>	>	>	>	>
+	<	<	<	<	<	>	>	>
-	<	<	<	<	<	>	>	>
\$	<	<	<	<	<	<	<	<

- ▶ Dacă vrem să analizăm $-2^2^x + 3 * 5 - 2 + 4$
- ▶ Transformăm în $\$<-<2>^<2>^<x>+<3>*<5>-<2>+<4>\$$

Algoritm

1. Punem pe stivă până la primul $>$
2. Când întâlnim $>$ scoatem din stivă până la $<$, și evaluăm
 - ▶ Am scos din stivă $< V1 \circ V2 >$
 - ▶ unde V -urile sunt valori deja obținute
 - ▶ Calculăm valoarea nouă V (arbore de parsare, număr)
3. Vedem relația dintre operatorul de pe stivă și cel din șir
 - ▶ dacă $<$, punem $< V$ pe stivă și mergem la (1)
 - ▶ dacă $>$, punem $V >$ pe stivă și mergem la (2)
 - ▶ dacă $=$ (aceeași producție), punem V pe stivă și mergem la (1)

Până când avem doar $\$$ în șir și operatorul rămas în stivă e $\$$

Exemplu

▶	$\$ < - < 2 > ^ < 2 > ^ < x > + < 3 > * < 5 > - < 2 > + < 4 > \$$		
▶	$^ < 2 > ^ < x > + < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 >$	
▶	$^ < 2 > ^ < x > + < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2$	$(0 - < ^)$
▶	$^ < x > + < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 ^ < 2 >$	
▶	$^ < x > + < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 ^ < 2$	$(^ < ^)$
▶	$+ < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 ^ < 2 ^ < x >$	
▶	$+ < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 ^ < 2 ^ x >$	$(^ > +)$
▶	$+ < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - < 2 ^ (2 ^ x) >$	$(^ > +)$
▶	$+ < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < - (2 ^ (2 ^ x)) >$	$(0 - > +)$
▶	$+ < 3 > * < 5 > - < 2 > + < 4 > \$$	$\$ < (- (2 ^ (2 ^ x)))$	$(\$ < +)$

Algoritm

1. Punem pe stivă până la primul >
2. Când întâlnim > scoatem din stivă până la <, și evaluăm
3. Vedem relația dintre operatorul de pe stivă și cel din șir
 - ▶ dacă <, punem < apoi valoarea pe stivă și mergem la (1)
 - ▶ dacă >, punem valoarea, apoi > pe stivă și mergem la (2)

Exemplu

- ▶ $+<3>*<5>-<2>+<4>\$ \$<(-(2^{(2^x)}))>$
- ▶ $*<5>-<2>+<4>\$ \$<(-(2^{(2^x)}))+<3>$
- ▶ $*<5>-<2>+<4>\$ \$<(-(2^{(2^x)}))+<3$ (+ < *)
- ▶ $-<2>+<4>\$ \$<(-(2^{(2^x)}))+<3*<5>$
- ▶ $-<2>+<4>\$ \$<(-(2^{(2^x)}))+<3*5>$ (* > -)
- ▶ $-<2>+<4>\$ \$<(-(2^{(2^x)}))+(3*5)>$ (+ > -)
- ▶ $-<2>+<4>\$ \$<((- (2^{(2^x)}))+(3*5))>$ (\$ < -)
- ▶ $+<4>\$ \$<((- (2^{(2^x)}))+(3*5))-<2>$
- ▶ $+<4>\$ \$<((- (2^{(2^x)}))+(3*5))-2>$ (- > +)
- ▶ $+<4>\$ \$<(((- (2^{(2^x)}))+(3*5))-2)$ (\$ < +)

Algoritm

1. Punem pe stivă până la primul >
2. Când întâlnim > scoatem din stivă până la <, și evaluăm
3. Vedem relația dintre operatorul de pe stivă și cel din șir
 - ▶ dacă <, punem < apoi valoarea pe stivă și mergem la (1)
 - ▶ dacă >, punem valoarea, apoi > pe stivă și mergem la (2)

Exemplu

- ▶ $+ < 4 > \$ \$ < (((-(2^{(2^x)})))+(3*5))-2)$
- ▶ $\$ \$ < (((-(2^{(2^x)})))+(3*5))-2) + < 4 >$
- ▶ $\$ \$ < (((-(2^{(2^x)})))+(3*5))-2) + 4 >$ (+ > \$)
- ▶ $\$ \$ < (((-(2^{(2^x)})))+(3*5))-2) + 4)$ (gata)

Algoritm

1. Punem pe stivă până la primul >
 2. Când întâlnim > scoatem din stivă până la <, și evaluăm
 3. Vedem relația dintre operatorul de pe stivă și cel din șir
 - ▶ dacă <, punem < apoi valoarea pe stivă și mergem la (1)
 - ▶ dacă >, punem valoarea, apoi > pe stivă și mergem la (2)
- Până când avem doar \$ în șir și operatorul rămas în stivă e \$

Surse

- ▶ Gatevidyalay
- ▶ Wikipedia
- ▶ Text.Parsec.Expr