

### 3.4. ГРАМАТИЧНИЙ АНАЛІЗ РЯДКІВ

Одним з основних результатів роботи синтаксичного аналізатора є формування дерева граматичного розбору, що є основною вхідною інформацією для складальника вихідного рядка компілятора.

Існує два методи розбору – спадний і висхідний.

У *спадному (низхідному)* граматичному розборі побудова синтаксичного дерева починається з кореня дерева (тобто з виділеного символу) і продовжується вниз до листів, тобто до термінальних символів аналізованого рядка. Мета такого процесу: систематично породжувати правила мови доти, доки не буде отримане речення, що збігається з аналізованим рядком. При цьому, можливо, доведеться повертатися назад, якщо вибраний раніше ланцюжок виявився тупиковим. Щоб уникнути частих повернень назад, розроблені різноманітні варіанти проглядів вперед. Чим складніше граматики мови, тим більше «дистанція» прогляду.

Породження правила можна відобразити за допомогою особливого відношення  $\Rightarrow$ , для якого запис ' $x \Rightarrow y$ ' інтерпретується як «рядок ' $x$ ' породжує рядок ' $y$ '».

Нижче наводиться алгоритм граматичного розбору для рядка « $a1b := b0 + 1.9/z ;$ ». Враховуючі малий обсяг цього рядка та щоб не ускладнювати сприйняття алгоритму, при граматичному аналізі не будемо використовувати лексеми. Зліва наведені номери використаних правил основної граматики *G4*.

- 1)  $\langle \text{оператор присвоєння } AB \rangle := \langle \text{ідентифікатор} \rangle := \langle AB \rangle ;$
- 10)  $\langle \text{ідентифікатор} \rangle := \langle AB \rangle ; \Rightarrow \langle \text{ідентифікатор} \rangle \langle \text{літера} \rangle := \langle AB \rangle ;$
- 12)  $\langle \text{ідентифікатор} \rangle \langle \text{літера} \rangle := \langle AB \rangle ; \Rightarrow \langle \text{ідентифікатор} \rangle \mathbf{b} := \langle AB \rangle ;$
- 11)  $\langle \text{ідентифікатор} \rangle \mathbf{b} := \langle AB \rangle ; \Rightarrow \langle \text{ідентифікатор} \rangle \langle \text{цифра} \rangle \mathbf{b} := \langle AB \rangle ;$
- 39)  $\langle \text{ідентифікатор} \rangle \langle \text{цифра} \rangle \mathbf{b} := \langle AB \rangle ; \Rightarrow \langle \text{ідентифікатор} \rangle \mathbf{1b} := \langle AB \rangle ;$
- 9)  $\langle \text{ідентифікатор} \rangle \mathbf{1b} := \langle AB \rangle ; \Rightarrow \langle \text{літера} \rangle \mathbf{1b} := \langle AB \rangle ;$

## Структури та організація даних в ЕОМ

---

- 12) <літера>**1b**:=<AB>;=>**a1b**:=<AB>;
- 3) **a1b**:=<AB>;=>**a1b**:=<AB><знак додавання/віднімання>  
<складове>;
- 2) **a1b**:=<AB><знак додавання/віднімання><складове>;  
=>**a1b**:=<складове><знак додавання/віднімання>  
<складове>;
- 4) **a1b**:=<складове><знак додавання/віднімання><складове>;  
=>**a1b**:=<множник><знак додавання/віднімання><складове>;
- 6) **a1b**:=<множник><знак додавання/віднімання><складове>;  
=>**a1b**:=<ідентифікатор><знак додавання/віднімання>  
<складове>;
- 11) **a1b**:=<ідентифікатор><знак додавання/віднімання>  
<складове>;=>**a1b**:=<ідентифікатор><цифра><знак  
додавання/віднімання><складове>;
- 38) **a1b**:=<ідентифікатор><цифра><знак додавання/віднімання>  
<складове>;=>**a1b**:=<ідентифікатор>**0**<знак додавання/  
віднімання><складове>;
- 9) **a1b**:=<ідентифікатор>**0**<знак додавання/віднімання>  
<складове>;=>**a1b**:=<літера>**0**<знак додавання/віднімання>  
<складове>;
- 12) **a1b**:=<літера>**0**<знак додавання/віднімання><складове>;  
=>**a1b**:=**b0**<знак додавання/віднімання><складове>;
- 53) **a1b**:=**b0**<знак додавання/віднімання><складове>;=>  
**a1b**:=**b0**+<складове>;
- 5) **a1b**:=**b0**+<складове>;=>**a1b**:=**b0**+<складове><знак  
множення/ділення><множник>;
- 4) **a1b**:=**b0**+<складове><знак множення/ділення><множник>;  
=>**a1b**:=**b0**+<множник><знак множення/ділення><множник>;
- 7) **a1b**:=**b0**+<множник><знак множення/ділення><множник>;  
=>**a1b**:=**b0**+<число><знак множення/ділення> <множник>;
- 48) **a1b**:=**b0**+<число><знак множення/ділення><множник>;  
=>**a1b**:=**b0**+<число без знака><знак множення/ділення>  
<множник>;
- 49) **a1b**:=**b0**+<число без знака><знак множення/ділення>  
<множник>;=>**a1b**:=**b0**+<ціле без знака>.<ціле без  
знака><знак множення/ділення><множник>;
- 51) **a1b**:=**b0**+<ціле без знака>.<ціле без знака><знак  
множення/ділення><множник>;=>**a1b**:=**b0**+ <цифра>.<ціле  
без знака><знак множення/ділення><множник>;

- 47)  $a1b := b0 + \langle \text{цифра} \rangle \cdot \langle \text{ціле без знака} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ; \Rightarrow a1b := b0 + 1 \cdot \langle \text{ціле без знака} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ;$
- 51)  $a1b := b0 + 1 \cdot \langle \text{ціле без знака} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ; \Rightarrow a1b := b0 + 1 \cdot \langle \text{цифра} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ;$
- 38)  $a1b := b0 + 1 \cdot \langle \text{цифра} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ; \Rightarrow a1b := b0 + 1 \cdot 9 \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ;$
- 56)  $a1b := b0 + 1 \cdot 9 \langle \text{знак множення/ділення} \rangle \langle \text{множник} \rangle ; \Rightarrow a1b := b0 + 1 \cdot 9 / \langle \text{множник} \rangle ;$
- 6)  $a1b := b0 + 1 \cdot 9 / \langle \text{множник} \rangle ; \Rightarrow a1b := b0 + 1 \cdot 9 / \langle \text{ідентифікатор} \rangle ;$
- 9)  $a1b := b0 + 1 \cdot 9 / \langle \text{ідентифікатор} \rangle ; \Rightarrow a1b := b0 + 1 \cdot 9 / \langle \text{літера} \rangle ;$
- 37)  $a1b := b0 + 1 \cdot 9 / \langle \text{літера} \rangle ; \Rightarrow a1b := b0 + 1 \cdot 9 / z ;$

Таким чином, послідовність застосованих правил при спадному розборі зліва направо наступна:  $R = \{1, 10, 12, 11, 39, 9, 12, 3, 2, 4, 6, 11, 38, 9, 12, 53, 5, 4, 7, 48, 49, 51, 38, 56, 6, 9, 37\}$ . Цей кортеж є вхідною інформацією для процесора складання вихідного рядка при роботі компілятора (рис. 3.4).

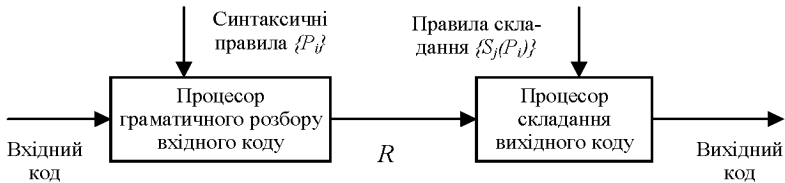


Рис. 3.4. Узагальнена схема роботи компілятора

При *підіймальному (висхідному)* граматичному розборі побудова синтаксичного дерева починається з «листів» (висячих вершин синтаксичного графа, якими є нетермінальні символи граматики) і продовжується просуванням нагору до кореня. При такому алгоритмі запис « $x$ »  $\Rightarrow$  « $y$ » інтерпретується як « $y$ » зводиться до « $x$ ». Застосовуючи цю стратегію до того ж вхідного рядка « $a1b := b0 + 1 \cdot 9 / z ;$ » при тій же граматиці  $G4$ , отримаємо наступну послідовність правил.

- 12)  $a1b := b0 + 1 \cdot 9 / z ; \Rightarrow \langle \text{літера} \rangle 1b := b0 + 1 \cdot 9 / z ;$
- 9)  $\langle \text{літера} \rangle 1b := b0 + 1 \cdot 9 / z ; \Rightarrow \langle \text{ідентифікатор} \rangle 1b := b0 + 1 \cdot 9 / z ;$
- 39)  $\langle \text{ідентифікатор} \rangle 1b := b0 + 1 \cdot 9 / z ; \Rightarrow \langle \text{ідентифікатор} \rangle \langle \text{цифра} \rangle b := b0 + 1 \cdot 9 / z ;$



- 4) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<множник>/z;=><ідентифікатор>:=<АВ><знак додавання/  
віднімання><складове>/z;
- 56) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове>/z;=><ідентифікатор>:=<АВ><знак додавання/  
віднімання><складове><знак множення/ділення>z;
- 37) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множ/діл>z;=><ідентифікатор>:=  
<АВ><знак додавання/віднімання><складове><знак  
множення/ділення><літера>;
- 9) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множення/ділення><літера>;=>  
<ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множення/ділення><ідентифікатор>;
- 6) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множення/ділення><ідентифікатор>;  
=><ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множення/ділення><множник>;
- 5) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове><знак множення/ділення><множник>;=>  
<ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове>;
- 3) <ідентифікатор>:=<АВ><знак додавання/віднімання>  
<складове>;=><ідентифікатор>:=<АВ>;
- 1) <ідентифікатор>:=<АВ>;=><оператор присвоєння АВ>

Таким чином, послідовність застосованих правил при висхідному розборі зліва направо наступна:  $R=\{12, 9, 39, 11, 13, 10, 13, 9, 38, 11, 6, 4, 2, 3, 39, 51, 47, 51, 49, 7, 4, 56, 37, 9, 6, 5, 3, 1\}$ .

Крім напрямку розбору «зліва направо» у кожному з алгоритмів розбору: спадного та висхідного – можна застосовувати і зворотній напрям розбору «справа наліво». Між ними існують певні зв'язки. Якщо позначимо:

$R_{c\rightarrow}$  – кортеж (послідовність) правил при спадному розборі зліва направо;

$R_{c\leftarrow}$  – кортеж правил при спадному розборі справа наліво;

$R_{v\rightarrow}$  – кортеж правил при висхідному розборі зліва направо;

$R_{v\leftarrow}$  – кортеж правил при висхідному розборі справа наліво;

то  $R_{c\rightarrow}=\text{revers}(R_{v\leftarrow})$ , а  $R_{c\leftarrow}=\text{revers}(R_{v\rightarrow})$ , де  $\text{revers}$  – функція реверсу послідовності правил у кортежі, тобто останнє правило стає першим, друге від кінця – другим і так далі, а перше правило стає в

## Структури та організація даних в ЕОМ

---

новому кортежі першим. Ці твердження пропонується перевірити самостійно, але при цьому правила з лівою рекурсією треба замінити на правила з правою рекурсією. Для граматики  $G_4$  це правила №№ 3, 5, 10 і 52, які при напрямі розбору справа наліво будуть виглядати так:

- 3)  $\langle AB \rangle ::= \langle \text{складове} \rangle \langle \text{знак додавання/віднімання} \rangle \langle AB \rangle$
- 5)  $\langle \text{складове} \rangle ::= \langle \text{множник} \rangle \langle \text{знак множення/ділення} \rangle \langle \text{складове} \rangle$
- 10)  $\langle \text{ідентифікатор} \rangle ::= \langle \text{літера} \rangle \langle \text{ідентифікатор} \rangle$
- 52)  $\langle \text{ціле без знака} \rangle ::= \langle \text{цифра} \rangle \langle \text{ціле без знака} \rangle$