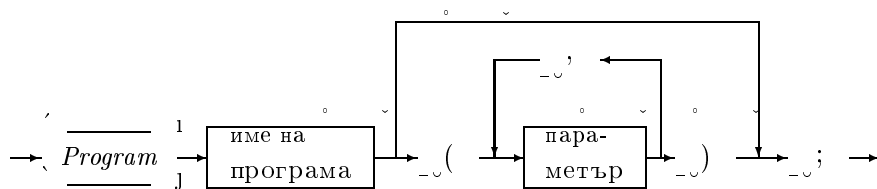


## 2.3 Пораждащи граматик

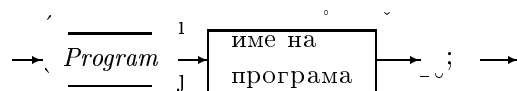
Повечето от описанията на алгоритмичните езици от високо ниво, съдържат един задължителен раздел, в който формално се описва синтаксисът на съответния език. Напоследък, за тази цел най-често се използват синтактични диаграми или дефиниции, посредством метазезикът наречен Бекус-Наурова форма.

Да разгледаме една синтактична диаграма, определяща понятието "заглавие на програма" в алгоритмичния език *Pascal* (Фиг. 2.3.1).



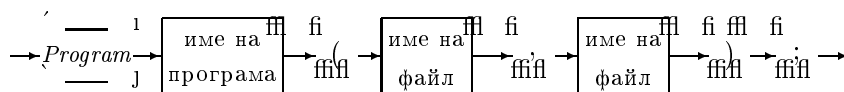
Фиг. 2.3.1

Всеки път по тази диаграма преминава през няколко "възла". Най-простият или най-краткият път, например е зададения на Фиг. 2.3.2.



Фиг. 2.3.2

По-сложен е пътят на Фиг. 2.3.3.



Фиг. 2.3.3

Не е трудно да се види, че съществуват безбройно много пътища по диаграмата, които съответствуват на всевъзможните определения на понятието "заглавие на програма". Разбира се, конкретните реализации на езика *Pascal*, налагат ограничения

върху броя на пътищата по дадената синтактична диаграма, но това за нашите разглеждания не е важно.

Равностойна дефиниция на същото понятие може да се даде и чрез Бекус-Науровата форма, която няма да разглеждаме сега.

На горната диаграма, възлите са разположени в овални или в правоъгълни блокове. Така например *Program*, е поставено в овален блок, а "име на програма" в правоъгълен. Разликата между двата блока е, че думите записани в овалните блокове са елементи (запазени думи или символи) от езика, а тези които са в правоъгълните блокове не са елементи на езика и те подлежат на дефиниране, чрез синтаксиса на езика.

Тази разлика между блоковете в синтактичните диаграми, е причина думите в овалните блокове да се наричат *терминални символи* на езика, а тези в правоъгълните – *нетерминални символи* на езика.

Ясно е, че всяка коректна програма представлява нетерминален символ на съответния език.

От друга страна, описанието на алгоритмичните езици включва и редица други изисквания, които не могат да се изразят, чрез синтактични диаграми (например изискването всички променливи да се опишат относно своя тип и др.).

Използвайки аналогията с по-горния пример, ще определим формален модел на граматика, която поражда съответен на нея формален език.

**Определение 2.3.1** *Под пораждаща граматика  $\Gamma$  разбираме наредена четворка  $\Gamma = \langle V, W, S, P \rangle$ , където  $V$  е крайно множество (азбука) от терминални символи,  $W$  – крайно множество (азбука) от нетерминални символи,  $S$  – начален символ на граматиката, който е и елемент на  $W$  и  $P$  е крайно множество от наредени двойки  $(\alpha, \beta)$ , за които  $\alpha, \beta \in (V \cup W)^*$  като  $\alpha$  има поне един нетерминален символ.*

Елементите на  $V$  най-често ще означаваме с малки латински букви  $a, b, \dots$  (евентуално с индекси  $a_1, a_2, \dots$ ), докато нетер-

миналните символи от  $W$  най-често ще означаваме с главните латински букви  $A, B, \dots$  (евентуално с индекси  $A_1, A_2, \dots$ ).

По-нататък, за избягване на двусмислие при пораждање на думи, ще предполагаме  $V \cap W = \emptyset$ .

Прието е елементите на  $P$  да се наричат още *правила* или *продукции* на граматиката  $\Gamma$ . Също така, ако  $(\alpha, \beta) \in P$  това често се означава с  $\alpha \rightarrow \beta$ , като символът " $\rightarrow$ " не е нито във  $V$ , нито в  $W$ .

Нека  $\mu$  и  $\nu$  са две думи от  $(V \cup W)^*$ . Ще казваме, че  $\mu$  се *извежда непосредствено от*  $\nu$  в граматиката  $\Gamma = \langle V, W, S, P \rangle$ , ако съществуват думи  $\alpha_1, \alpha_2 \in (V \cup W)^*$  и правило  $\alpha \rightarrow \beta$  в  $P$  така, че

$$\nu = \alpha_1 \alpha \alpha_2 \text{ и } \mu = \alpha_1 \beta \alpha_2.$$

Когато  $\mu$  се извежда от  $\nu$  в  $\Gamma$  ще пишем  $\nu \stackrel{\Gamma}{\vdash} \mu$  или само  $\nu \vdash \mu$ , ако граматиката  $\Gamma$  се подразбира.

Ако  $\omega_1, \omega_2, \dots, \omega_n$  са думи над  $V \cup W$ , за които

$$\omega_1 \stackrel{\Gamma}{\vdash} \omega_2 \stackrel{\Gamma}{\vdash} \dots \stackrel{\Gamma}{\vdash} \omega_n,$$

ще казваме, че тази редица от думи е *извод на*  $\omega_n$  *от*  $\omega_1$  *в*  $\Gamma$ .

Често това ще означаваме с  $\omega_1 \stackrel{\Gamma}{\models} \omega_n$  или само  $\omega_1 \models \omega_n$ .

Броят на непосредствените изводи  $\omega_i \vdash \omega_{i+1}$  в горната редица, ще наричаме *дължина на извода*  $\omega_1 \models \omega_n$ .

Всяка дума  $\omega \in (V \cup W)^*$ , за която  $S \models \omega$  се нарича *сентенциална форма* на граматиката  $\Gamma$ .

От казаното следва, че ако в  $P$  има правило  $\alpha \rightarrow \beta$ , то във всяка дума от вида  $\alpha_1 \alpha \alpha_2$  поддумата  $\alpha$  може да се замени с  $\beta$  и по този начин да се получи думата  $\alpha_1 \beta \alpha_2$ .

**Определение 2.3.2** Множеството от всички думи  $\omega$  над  $V$ , за които  $S \stackrel{\Gamma}{\models} \omega$  се нарича **формален език над  $V$** , породен от граматиката  $\Gamma$ .

Често този език се означава с  $L(\Gamma)$ .

Ще казваме, че граматиките  $\Gamma_1$  и  $\Gamma_2$  са *еквивалентни*, ако  $L(\Gamma_1) = L(\Gamma_2)$ .

Често ще използваме означенията  $\alpha \rightarrow \beta|\gamma|\delta|\dots|\kappa$  за групата от правила  $\alpha \rightarrow \beta, \alpha \rightarrow \gamma, \alpha \rightarrow \delta, \dots, \alpha \rightarrow \kappa$ .

**Пример 2.3.1** Нека  $\Gamma_1 = \langle V, W, S, P \rangle$  е пораздаща граматика, за която  $V = \{a, b\}, W = \{S, A, B\}$ ,  $S$  е начален символ и  $P$  се състои от правилата

$$S \rightarrow A|B, \quad A \rightarrow aA|a, \quad B \rightarrow bB|b.$$

Не е трудно да се покаже, че от тази граматика могат да се породят всички думи от вида  $\alpha = a^n, n \geq 1$  или от вида  $\beta = b^n, n \geq 1$ .

**Пример 2.3.2** Нека  $\Gamma_2$  има същите азбуки  $V$  и  $W$ , както в предходния пример, а  $P$  се състои от правилата

$$S \rightarrow aA|bB|a|b, \quad A \rightarrow aA|a, \quad B \rightarrow bB|b.$$

Лесно се показва, че  $L(\Gamma_1) = L(\Gamma_2)$ . Следователно  $\Gamma_1$  и  $\Gamma_2$  са еквивалентни пораздащи граматики.

**Пример 2.3.3** Нека  $\Gamma_3 = \langle V, W, S, P \rangle$  е граматика, за която  $V = \{a, b\}, W = \{S\}$  и  $P$  се състои от правилата

$$S \rightarrow aSb|ab.$$

В тази граматика думата  $a^3b^3$  може да се породи, чрез извода:

$$S \vdash aSb \vdash aaSbb \vdash aaabbb.$$

Също така лесно се показва, че

$$L(\Gamma_3) = \{a^n b^n \mid n \geq 1\}.$$

Ясно е, че  $\Gamma_3$  не е еквивалентна на  $\Gamma_1$  и  $\Gamma_2$ , но тя се различава от тях и по това, че началният символ  $S$  се среща и в дясната страна на правилата в  $\Gamma_3$ . Съществува граматика  $\Gamma'$ , еквивалентна на  $\Gamma_3$ , за която в дясната страна на правилата, началният символ не се съдържа.

**Лема 2.3.1** *Нека  $\Gamma = \langle V, W, S, P \rangle$  е граматика, която има правила, в които началният символ се съдържа в дясната им страна. Съществува граматика  $\Gamma'$ , еквивалентна на  $\Gamma$  и в дясната страна на правилата на  $\Gamma'$ , началният символ на  $\Gamma'$  не се съдържа.*

**Доказателство:** Нека  $\Gamma' = \langle V', W', S', P' \rangle$  е граматика, за която:

- (i)  $V' = V$
- (ii)  $W' = W \cup S'$ , където  $S' \notin V \cup W$
- (iii)  $S'$  е начален символ на  $\Gamma'$
- (iv) в  $P'$  са всички правила на  $P$  и за всяко правило  $S \rightarrow \alpha$  от  $P$  се добавя правилото  $S' \rightarrow \alpha$ .

Очевидно в  $P'$  всички правила са с дясна страна, която не съдържа  $S'$ .

Да се убедим, че  $L(\Gamma) = L(\Gamma')$ .

Нека  $\omega \in L(\Gamma)$ . Следователно съществува извод  $S \stackrel{\Gamma}{\vdash} \omega$ . Ако дължината на този извод е 1, то в  $P$  има правило  $S \rightarrow \omega$  и следователно в  $P'$  има правило  $S' \rightarrow \omega$ , т.е.  $\omega \in L(\Gamma')$ . Ако дължината на извода  $S \stackrel{\Gamma}{\vdash} \omega$  е по-голям от единица, то имаме

$$S \stackrel{\Gamma}{\vdash} \alpha \stackrel{\Gamma}{\vdash} \omega,$$

където  $S \rightarrow \alpha$  е правило в  $P$ . Следователно в  $P'$  съществува правило  $S' \rightarrow \alpha$ . Тъй като всички правила в  $P$  са правила на  $P'$ , индуктивно следва, че

$$S' \stackrel{\Gamma'}{\vdash} \alpha \stackrel{\Gamma'}{\vdash} \omega, \text{ т.е. } \omega \in L(\Gamma').$$

Нека сега  $\omega \in L(\Gamma')$ . Следователно съществува изводът  $S' \stackrel{\Gamma'}{\vdash} \omega$ . Ако този извод има дължина 1, то  $S' \rightarrow \omega$  е правило в  $P'$  и следователно  $S \rightarrow \omega$  е правило в  $P$ , т.е.  $\omega \in L(\Gamma)$ . Ако изводът  $S' \stackrel{\Gamma'}{\vdash} \omega$  има дължина по-голяма от 1 можем да запишем

$$S' \stackrel{\Gamma'}{\vdash} \alpha \stackrel{\Gamma'}{\vdash} \omega,$$

където  $S \rightarrow \alpha$  е правило в  $P'$ . Следователно  $S \rightarrow \alpha$  е правило на  $P$ . Но тогава  $S'$  не участва в нито една дума от извода  $\alpha \stackrel{\Gamma}{=} \omega$ . Ето защо, този извод може да се осъществи и с правилата в  $P$ , т.е.

$$S \stackrel{\Gamma}{\vdash} \alpha \stackrel{\Gamma}{=} \omega$$

е извод в  $\Gamma$ . Следователно  $\omega \in L(\Gamma)$ .  $\square$ .

За граматиката  $\Gamma_3$  от Пример 2.3.3, получаваме следната граматика  $\Gamma'_3$  еквивалентна на нея:

$$\Gamma'_3 = \langle \{a, b\}, \{S, S'\}, S', \{S' \rightarrow aSb|ab, S \rightarrow aSb|ab\} \rangle,$$

която не съдържа началния символ  $S'$  в дясната страна на своите правила.

Оказва се, че всички пораждащи граматики могат да бъдат класифицирани по отношение на структурата на техните правила.

Пораждащи граматики, които имат правила от вида

$$\alpha A \beta \rightarrow \omega, \quad \alpha, \beta, \omega \in (V \cup W)^*, \quad A \in W$$

се наричат граматики от *общ вид* или *от тип 0*.

Пораждащи граматики, на които правилата са от вида

$$\alpha A \beta \rightarrow \alpha \omega \beta, \quad \omega \neq \varepsilon, \quad \alpha, \beta, \omega \in (V \cup W)^*, \quad A \in W$$

се наричат *контексни граматики* или *от тип 1*.

Очевидно  $A$  може да се замества с  $\omega$  само, ако има дадения контекст  $\alpha - \beta$ . Често  $\alpha$  се нарича ляв, а  $\beta$  – десен контекст на  $A$ . Тъй като  $|\alpha \omega \beta| \geq 1$ , то празната дума не се поражда в тези граматики. Ето защо, когато е необходимо ще добавяме и правилото  $S \rightarrow \varepsilon$  и получената граматика отново ще смятаме за контекстна. Правилото  $S \rightarrow \varepsilon$  може да бъде използвано само еднократно в изводите и при това само за извода  $S \models \varepsilon$ . Ето защо това правило не влияе върху останалите думи, различни от  $\varepsilon$  и породени от граматиката.

**Пример 2.3.4** Граматиката

$$\Gamma_4 = \langle \{a, b, c\}, \{S, A, C\}, S, \{S \rightarrow aSAC|abc, \\ cA \rightarrow Ac, bA \rightarrow bb, C \rightarrow c\} \rangle$$

е контекстна и тя поражда всички думи от вида

$$\alpha = a^n b^n c^n \text{ за } n \geq 1.$$

Пораждащи граматиките, на които правилата са от вида

$$A \rightarrow \omega, \quad A \in W, \quad \omega \in (V \cup W)^+$$

се наричат *безконтекстни* или *от тип 2*. Както в случая на контекстна граматика, може да се добави правило  $S \rightarrow \varepsilon$ , което се използва само за пораждаване на празната дума.

**Пример 2.3.5** Граматиката

$$\Gamma_5 = \langle \{a, b\}, \{S, A, B\}, S, \{S \rightarrow AB, A \rightarrow aA|a, B \rightarrow bB|b\} \rangle$$

е безконтекстна и тя поражда езика

$$L(\Gamma_5) = \{a^n b^m \mid n, m \geq 1\}.$$

Пораждащи граматиките, чиито правила са от вида

$$A \rightarrow aB|a, \quad A, B \in W, \quad a \in V$$

се наричат *автоматни* или *от тип 3*. И тук е допустимо да се причисли правилото  $S \rightarrow \varepsilon$ , което може да се използва само за пораждането на празната дума.

**Пример 2.3.6** Граматиката

$$\Gamma_6 = \langle \{a, b\}, \{S, A, B\}, S, \\ \{S \rightarrow aA|bB|a|b, A \rightarrow aA|a, B \rightarrow bB|b, S \rightarrow \varepsilon\} \rangle$$

е автоматна и поражда езика

$$L(\Gamma_6) = \{a^n \mid n \geq 0\} \cup \{b^n \mid n \geq 0\}.$$

Ясно е, че всяка автоматна граматика е безконтекстна, всяка безконтекстна граматика е контекстна и всяка контекстна граматика е граматика от общ вид. Налице са следните включвания:  $\{\text{граматики от тип 3}\} \subset \{\text{граматики от тип 2}\} \subset \{\text{граматики от тип 1}\} \subset \{\text{граматики от тип 0}\}$ .

Тези включвания, често се наричат *йерархия на Чомски* за поразждащите граматики.

Езиците породени от граматика от тип  $i$  ( $i = 0, 1, 2, 3$ ) се наричат *езици от тип  $i$*  за  $i = 0, 1, 2, 3$  или още *езици от общ вид, контекстни, безконтекстни и автоматни*, съответно.

Аналогично на поразждащите ги граматики и породените езици образуват йерархия на Чомски.

Без особени затруднения, може да се докаже следната лема.

**Лема 2.3.2** *Ако  $L$  е формален език от тип  $i$  ( $i = 0, 1, 2, 3$ ), то и езиците  $L \cup \{\varepsilon\}$  и  $L \setminus \{\varepsilon\}$  са от същия тип  $i$ .*  $\square$

### З а д а ч и

1. Да се определи езикът, който се поразжда от граматиката  $\Gamma$ , ако

- а)  $\Gamma = \langle \{a, b\}, \{S, A\}, S, \{S \rightarrow \varepsilon | A, A \rightarrow Ab | aA | a | b\} \rangle$ ;
- б)  $\Gamma = \langle \{a, b, c\}, \{S\}, S, \{S \rightarrow aSa | bSb | c\} \rangle$ ;
- в)  $\Gamma = \langle \{a, b\}, \{S, A, B\}, S, \{S \rightarrow A, A \rightarrow aAb | B, B \rightarrow bB | b\} \rangle$ .

2. Да се опише граматика, която поразжда следните езици:

- а)  $L = \{a^n b^{2n} \mid n \geq 1\}$ ;
- б)  $L = \{a^n b^2 a^n \mid n \geq 1\}$ ;
- в)  $L = \{a, ab, abb, abbb\}$ ;
- г)  $L = \{\alpha \in \{a, b\}^* \mid \alpha \text{ няма за поддума } a^2\}$ ;
- д)  $L = \{\alpha \in \{a, b\}^* \mid \text{в } \alpha \text{ броя на появяванията на буквата } a \text{ е два пъти по-голям от броя на появяванията на } b\}$ ;
- е)  $L = \{\alpha \in \{a, b\}^* \mid \alpha = \overline{\alpha}, \text{ където } \overline{\alpha} \text{ е думата } \alpha \text{ четена отдясно наляво}\}$ ;
- ж)  $L = \{\alpha \in \{0, 1\}^* \mid \alpha \text{ започва с две нули}\}$ ;
- з)  $L = \{\alpha \in \{0, 1\}^* \mid \alpha \text{ има нечетен брой единици}\}$ ;



и)  $L = \{\alpha \in \{0, 1\}^* \mid \alpha \text{ съдържа точно една нула}\};$

к)  $L = \{\alpha \in \{0, 1\}^* \mid \alpha \text{ не завършва с две единици}\}.$

3. Да се докаже, че една граматика  $\Gamma = \langle V, W, S, P \rangle$ , непораждаща  $\varepsilon$  е автоматна точно тогава, когато съществува еквивалентна на нея граматика, чиито правила са от вида

$$A \rightarrow Ba|a, \quad A, B \in W, \quad a \in V.$$