

2. Formālās gramatikas

Nodaļas saturs

2.	Formālās gramatikas	2-1
2.1.	Regulāras izteiksmes	2-2
2.1.1.	Valoda un tās sintakses apraksts	2-2
2.1.2.	Regulāras izteiksmes definēšana	2-3
2.1.3.	Regulāro izteiksmju iespējas	2-4
2.2.	Gramatikas	2-4
2.2.1.	Gramatikas definēšana	2-4
2.2.2.	Gramatiku piemēri	2-5
2.2.3.	Bezkonteksta gramatikas	2-6
2.2.4.	Konteksta atkarīgās gramatikas	2-7
2.3.	Gramatiku klases	2-9
2.4.	Vingrinājumi	2-9

Nodaļā ievietotie attēli, programmu piemēri un tabulas

Att. 2-1.	Valoda #1	2-2
Att. 2-2.	Valoda #2	2-2
Att. 2-3.	Valoda #3	2-2
Att. 2-4.	Valoda #4	2-3
Att. 2-5.	Regulāra izteiksme valodai #4 ($\{x^m y^n \mid m, n \geq 0\}$)	2-3
Att. 2-6.	Divas alternatīvas regulāras izteiksmes valodai #3 ($\{x^m y^n \mid m, n > 0\}$)	2-4
Att. 2-7.	Regulāra izteiksme valodai #5 (simbolu a virkne vai simbolu b virkne)	2-4
Att. 2-8.	Regulāra izteiksme valodai #6 (vārdi no a un b jebkurā kombinācijā)	2-4
Att. 2-9.	Gramatika Γ_4 (valodai $\{x^m y^n \mid m, n \geq 0\}$)	2-5
Att. 2-10.	Gramatika Γ_2 (valodai $\{x^m y^n \mid m, n > 0\}$)	2-5
Att. 2-11.	Gramatika Γ_2 saīsinātajā variantā	2-6
Att. 2-12.	Gramatika Γ_7 (valodai $\{a^m b^n c^{m+n} \mid m, n > 0\}$)	2-7
Att. 2-13.	Gramatika Γ_7 , alternatīvs variants	2-7
Att. 2-14.	Gramatika Γ_8 (valodai $\{a^n b^n c^n \mid n \geq 0\}$)	2-8
Att. 2-15.	Gramatika Γ_9 (valodai $\{a^n \mid n \text{ ir nenegatīva divnieku pakāpe}\}$)	2-8
Tab. 2-1.	Regulāru izteiksmju operatoru un gramatiku produkciju konstrukciju saistība	2-6
Tab. 2-2.	Gramatiku klasifikācija	2-9

Valoda (language) ir vārdu kopums, kā arī šo vārdu nozīme vai plānotais efekts.

Gramatika (grammar) ir elementu un noteikumu kopums, kas nosaka valodai piederošo vārdu (simbolu virkņu) kopumu.

Gramatikas formāla definīcija tiks apskatīta zemāk.

Gramatikas šajā mācību materiālā iekļautas tāpēc, ka to veidošana daudzos gadījumos atgādina Prolog programmas veidošanu un prasa līdzīgas iemaņas, tādējādi ilustrējot to, ka Prolog programmēšanas “greizā” domāšana (salīdzinot ar tradicionālo programmēšanu) ir pārstāvēts arī citās sfērās un ir uzskatāms par pietiekoši dabisku informācijas reprezentācijas veidu.

2.1. Regulāras izteiksmes

Regulāras izteiksmes ir salīdzinoši vienkāršu valodu sintakses apraksta veids, tādu valodu, kuru pierakstam nav obligāti nepieciešama gramatika. Salīdzinot ar gramatiku, regulāra izteiksme ir īsāk pierakstāma un vieglāk uztverama. Kā vēlāk varēs redzēt, jebkuru valodu, kuru var aprakstīt ar regulāru izteiksmi, var aprakstīt ar gramatiku, bet ne tuvu ne otrādi.

2.1.1. Valoda un tās sintakses apraksts

Valodas sintakse nosaka visas simbolu virknes (vārdus), kas pieder dotajai valodai. Ja valoda sastāv no galīga skaita vārdu, tad sintakses definēšana nav pārāk sarežģīta, jo to var veikt, vienkārši pārskaitot visus vārdus, kuru gan varētu būt pietiekoši daudz.

Tā kā mūs parasti interesē valodas ar bezgalīgu vārdu kopu, tad būtu jāatrod veids, kā reprezentēt bezgalīgu daudzumu vārdu galīgā veidā.

Piemēram, apskatīsim valodu, kas sastāv no patvaļīga skaita simbolu x .

Att. 2-1. Valoda #1.

$\{x^n \mid n > 0\}$

Šeit zīme “ $|$ ” jālasa, kā “kur”, n – vesels skaitlis, bet reizināšanu (kas izriet no kāpināšanas) būtu jāsaprot kā konkatenāciju.

Šīs valodas vārdu piemēri:

x
 $xxxx$

Nākošā valoda sastāv no vārdiem, kas sastāv no viena vai vairākiem simboliem x , pēc kuriem seko tāds pats skaits simbolu y .

Att. 2-2. Valoda #2.

$\{x^n y^n \mid n > 0\}$

Šīs valodas vārdu piemēri:

xy
 $xxxyyy$
 $xxxxyyyy$

Ja x un y skaitam nav jāsakrīt, tad valodu pieraksta šādi:

Att. 2-3. Valoda #3.

$\{x^m y^n \mid m, n > 0\}$

Šai valodai pieder visi vārdi, kas pieder valodai #2, bet arī šādi:

$xyyy$
 $xxxxxyy$

Iespējama arī šāda valoda, kurai, salīdzinot ar valodu #3, x un y sastopamība nav obligāta (t.i., tie var vispār nebūt pārstāvēti vārdā):

Att. 2-4. Valoda #4.

$\{ x^m y^n \mid m, n \geq 0 \}$

Šai valodai pieder visi vārdi, kas pieder valodai #3, bet arī šādi:

xxx

yyyy

ϵ epsilon – tukšais vārds, ja $m=0$ un $n=0$

Tukšajam vārdam ϵ (epsilon), kā tālāk būs redzams, ir liela nozīme valodu definēšanā.

2.1.2. Regulāras izteiksmes definēšana

Visas augstāk aprakstītās valodas ir ērti aprakstāmas arī ar regulāru izteiksmju palīdzību.

Lai formāli definētu regulāru izteiksmi, vispirms ir jāievieš alfabēta jēdziens.

Alfabēts (*alphabet*) ir kādas valodas simbolu kopa.

Alfabētu piemēri:

$\{x\}$

$\{x, y\}$

$\{0, 1\}$

$\{a..z\}$

$\{0..9\}$

$\{a..z, A..Z, 0..9, _\}$

Regulāra izteiksme (*regular expression*) ir konstrukcija, kura nosaka kādas valodas vārdu kopu fiksētā alfabētā A.

1. Ja A ir alfabēts, tad pie regulārām izteiksmēm var pieskaitīt šādus elementus:

ϵ tukšais vārds

$a \in A$ jebkurš A elements

2. Bez tam, ja P un Q ir regulāras izteiksmes, par regulārām izteiksmēm uzskatāmas arī šādas konstrukcijas:

PQ Q seko pēc P

$P \mid Q$ P vai Q

P^* Neviena vai vairākas reizes atkārtots P

(P)

P^+ Viena vai vairākas reizes atkārtots P (+ oficiāli neskaitās regulāras izteiksmes operators, jo to var aizvietot ar *, šeit: PP^*)

Kā pirmo piemēru apskatīsim valodu #4 (sk. arī Att. 2-4, kurā ir vārdi, kas sākas ar burtiem x un turpinās ar burtiem y (abi var būt skaitā 0):

Att. 2-5. Regulāra izteiksme valodai #4 ($\{ x^m y^n \mid m, n \geq 0 \}$).

$x^* y^*$

Tālāk divi varianti valodas #3 aprakstam, kurā ir vārdi, kas sākas ar burtiem x un turpinās ar burtiem y (abiem jābūt skaitā vismaz 1):

Att. 2-6. Divas alternatīvas regulāras izteiksmes valodai #3 ($\{x^m y^n \mid m, n > 0\}$).

$x^+ y^+$
 $x^* x y^* y$

Nākošā valoda (#5) satur vārdus, kas sastāv vai nu no burtiem a, vai nu no burtiem b:

Att. 2-7. Regulāra izteiksme valodai #5 (simbolu a virkne vai simbolu b virkne).

$a^+ \mid b^+$

Valodas #5 vārdu piemēri:

a
aaa
bb

Valoda #6 satur vārdus, kas sastāv no burtiem a un b jebkurā secībā un kombinācijā, ieskaitot tukšo vārdu:

Att. 2-8. Regulāra izteiksme valodai #6 (vārdi no a un b jebkurā kombinācijā).

$(a \mid b)^*$

Valodas #6 vārdu piemēri:

ϵ
aaa
bb
abba

2.1.3. Regulāro izteiksmju iespējas

Ar regulārajām izteiksmēm var aprakstīt tikai ļoti vienkāršas valodas – regulārās (jeb 3. tipa) valodas. Viena no vienkāršām valodām, kuru **nevar** aprakstīt ar regulārām izteiksmēm, ir valoda #2:

$\{x^n y^n \mid n > 0\}$

kas sastāv no vārdiem, kas sastāv no viena vai vairākiem simboliem x, pēc kuriem seko **tāds pats skaits** simbolu y, jo ar regulāru izteiksmju palīdzību nav iespējams sinhronizēt dažādu elementu skaitu vārdos.

Savukārt, izmantojot bezkonteksta gramatikas, kas aprakstītas nākošajā nodaļā, šo valodu ir iespējams aprakstīt.

2.2. Gramatikas

2.2.1. Gramatikas definēšana

Gramatikas vispārīgā gadījumā dod iespēju aprakstīt sarežģītākas valodas nekā to var izdarīt ar regulārām izteiksmēm.

Gramatiku definē kā četrinieku:

$\Gamma = (V_T, V_N, P, S)$

kur

V_T terminālo simbolu (jeb termināļu) kopa – simbolu alfabēts, no kuriem sastāv

	aprakstāmās valodas vārdi
V_N	neterminālo simbolu (jeb netermināļu) kopa – simbolu alfabēts, kuri tiek izmantoti kā palīglīdzeklis valodas aprakstīšanā
P	produkciju (noteikumu) kopa
S	gramatikas sākuma simbols jeb aksioma

Kopām V_T un V_N nav kopīgu simbolu (kāds simbols nevar reizē būt gan terminālis, gan neterminālis):

$$V_T \cap V_N = \emptyset$$

Terminālo un neterminālo simbolu apvienojumu apzīmē ar V :

$$V = V_T \cup V_N$$

Katrs produkciju kopas P elements ir pāris (α, β) , kur α – produkcijas kreisā puse, bet β – produkcijas labā puse, ko pieraksta šādi:

$$\alpha \rightarrow \beta$$

α pieder pie V^+ (viens vai vairāki V simboli), bet β pieder pie V^* (neviens vai vairāki V simboli).

Gramatiku lieto, lai ģenerētu simbolu virknes – valodas vārdus, sākot ar aksiomu un pakāpeniski to nomainot, pielietojot produkcijas, tā, līdz beidzot tiek iegūta termināļu virkne.

Pie valodas pieder tie un tikai tie vārdi, kurus var izvest, izmantojot doto gramatiku.

2.2.2. Gramatiku piemēri

Kā pirmais piemērs tiks parādīts gramatikas pieraksts valodai #4 (salīdzināt ar Att. 2-5), kas jau tikusi aprakstīta ar regulāru izteiksmi.

Att. 2-9. Gramatika Γ_4 (valodai $\{x^m y^n \mid m, n \geq 0\}$).

$$\Gamma_4 = (\{x, y\}, \{S, X, Y\}, P_4, S)$$

P_4 :

1. $S \rightarrow XY$
2. $X \rightarrow \varepsilon$
3. $X \rightarrow xX$
4. $Y \rightarrow \varepsilon$
5. $Y \rightarrow yY$

Piemēram, vārdu

$xxxxyy$

gramatiku Γ_4 var ģenerēt šādā secībā (papildus dots numurs tai produkcijai, pēc kuras katrā solī notiek izvedums):

$$S \xRightarrow{1.} XY \xRightarrow{3.} xXY \xRightarrow{3.} xxXY \xRightarrow{3.} xxxXY \xRightarrow{2.} xxxY \xRightarrow{5.} xxxyY \xRightarrow{5.} xxxyyY \xRightarrow{4.} xxxyy$$

Ar šo gramatiku valodu #4 apraksta garāk nekā to var izdarīt ar regulāru izteiksmi, tomēr ir ļoti daudz valodu, kuras var aprakstīt tikai ar gramatiku, piemēram, jau pieminētā valoda #2 – valoda, kas sastāv no vārdiem, kur sākuma ir simbols x , beigās ir simboli y , turklāt x un y skaits sakrīt:

Att. 2-10. Gramatika Γ_2 (valodai $\{x^m y^n \mid m, n > 0\}$).

$$\Gamma_2 = (\{x, y\}, \{S\}, P_2, S)$$

P_2 :

1. $S \rightarrow xy$
2. $S \rightarrow xSy$

Piemēram, vārdu

$xyxy$

ar gramatiku Γ_2 var ģenerēt šādā secībā:

$$S \xRightarrow{2} xSy \xRightarrow{1} xyxy$$

Ja salīdzina regulāras izteiksmes ar gramatikām, tad kaut kādā nozīmē regulāra izteiksme ir gramatika, kurā ir tikai terminālie simboli un viena produkcija, turklāt produkcijai uzrādīta tikai labā puse, bez tam pierakstot produkciju bez terminālajiem simboliem drīkst lietot arī operatorus.

Lai kompensētu operatoru trūkumu ($*$, $+$, $|$), gramatikās izmanto šādas papildus konstrukcijas:

- iespējamās vairākas produkcijas ar vienādu kreiso pusi,
- produkcijas var definēt rekursīvi (labā puse var ietvert kreiso pusi).

Nākošā tabula parāda regulāru izteiksmju un gramatiku produkciju konstrukciju saistību:

Tab. 2-1. Regulāru izteiksmju operatoru un gramatiku produkciju konstrukciju saistība.

Regulāra izteiksme	Gramatika
$a \mid b$	1. $S \rightarrow a$ 2. $S \rightarrow b$
a^*	1. $S \rightarrow \epsilon$ 2. $S \rightarrow aS$
a^+	1. $S \rightarrow s$ 2. $S \rightarrow aS$

Regulārajās izteiksmēs pieejamo alternatīvās izvēles konstrukciju $Q \mid P$ var lietot arī gramatiku pierakstā, lai to saīsinātu. Tādējādi gramatika #2 saīsinātajā variantā izskatītos šādi:

Att. 2-11. Gramatika Γ_2 saīsinātajā variantā.

$$\Gamma_2 = (\{x, y\}, \{S\}, P_2, S)$$

$$P_2:$$

$$S \rightarrow xy \mid xSy$$

2.2.3. Bezkonteksta gramatikas

Valodu aprakstīšanai, ja vien iespējams, izmanto bezkonteksta gramatikas – tās ir daudz vieglāk realizēt nekā gramatikas, kuras nav bezkonteksta (varētu teikt, ka pārējām gramatikām vispārīgā gadījumā tas efektīvi nemaz nav izdarāms).

Bezkonteksta (*context free*) gramatika ir tāda gramatika, kuras visām produkcijām kreisajā pusē drīkst būt tikai viens neterminālais simbols.

Abas iepriekš apskatītās gramatikas Γ_2 un Γ_4 ir bezkonteksta gramatikas.

Vēl viens piemērs bezkonteksta gramatikai (vārdi sastāv no burtiem a, b, c, vispirms burti a, tad b un tikai tad c, turklāt burtu c skaits ir vienāds ar burtu a un burtu b skaitu summu, visiem 3 burtiem jābūt pārstāvētiem vārdā):

Att. 2-12. Gramatika Γ_7 (valodai $\{a^m b^n c^{m+n} \mid m, n > 0\}$).

$\Gamma_7 = (\{a, b, c\}, \{S, T, U\}, P_7, S)$

P_7 :

1. $S \rightarrow aTc$

2. $T \rightarrow aTc$

3. $T \rightarrow U$

4. $U \rightarrow bc$

5. $U \rightarrow bUc$

Valodas #7 vārdu piemēri:

abcc

aabccc

abbccc

aaaabbcccccc

Vienai un tai pašai valodai var uzrakstīt daudz alternatīvu gramatiku:

Att. 2-13. Gramatika Γ_7 , alternatīvs variants.

$\Gamma_7 = (\{a, b, c\}, \{S, T, U\}, P_7, S)$

P_7 :

1. $S \rightarrow aTc$

2. $T \rightarrow aTc$

3. $T \rightarrow bUc$

4. $U \rightarrow bUc$

5. $U \rightarrow \varepsilon$

Tomēr ir ļoti daudz pat ļoti vienkāršu valodu, kurām nevar uzrakstīt bezkonteksta gramatiku, lūk, piemērs:

Valoda #8 $\{a^n b^n c^n \mid n \geq 0\}$.

Katrs vārds sastāv no burtiem a, b, c (vispirms burti a, tad b un tikai tad c), un visi burtu veidi ir vienādā skaitā.

Valodas #8 vārdu piemēri:

abc

aabbcc

aaaabbbbcccc

2.2.4. Konteksta atkarīgās gramatikas

Konteksta atkarīgajām (*context sensitive*) gramatikām nav ierobežojuma, ka produkcijas kreisajā pusē drīkst būt tikai viens neterminālais simbols.

Vispārīgi runājot, konteksta atkarīgajām gramatikām (1. tipa gramatikām) ir ierobežojums, ka to produkciju kreisās puses nedrīkst būt garākas par labajām pusēm, citādi tās būtu pieskaitāmas pie 0. tipa gramatikām. Piemēram, sekojošās divas produkcijas ir pretrunā ar konteksta atkarīgas gramatikas nosacījumiem:

$AB \rightarrow C$

$A \rightarrow \varepsilon$

Tomēr šajā nodaļā tādas nianse netiks ņemtas vērā, un tiks apskatītas gramatikas, kuras vienkārši **nav bezkonteksta** gramatikas.

Konteksta atkarīgu gramatiku uzrakstīšana ir daudz sarežģītāka nekā vienkāršākām gramatikām un prasa daudz lielāku abstrakcijas līmeni.

Lūk, iepriekš apskatītās valodas #8 gramatikas piemērs:

Att. 2-14. Gramatika Γ_8 (valodai $\{a^n b^n c^n \mid n \geq 0\}$).

$\Gamma_8 = (\{a, b, c\}, \{S, T, U, B, C\}, P_8, S)$

P_8 :

1. $S \rightarrow aSBC$
2. $S \rightarrow T$
3. $CB \rightarrow BC$
4. $TB \rightarrow bT$
5. $T \rightarrow U$
6. $UC \rightarrow cU$
7. $UC \rightarrow c$

Piemēram, vārdu

$aaabbbcccc$

ar gramatiku Γ_8 var ģenerēt šādā secībā:

$$\begin{aligned} S &\Rightarrow aSBC \Rightarrow aaSBCBC \Rightarrow aaaSBCBCBC \Rightarrow aaaTBCBCBC \Rightarrow \\ &\Rightarrow aaaTBCCBC \Rightarrow aaaTBBCBCC \Rightarrow aaaTBBBCCC \Rightarrow \\ &\Rightarrow aaabTBCCCC \Rightarrow aaabbTBCCCC \Rightarrow aaabbbTCCCC \Rightarrow aaabbbUCCCC \Rightarrow \\ &\Rightarrow aaabbbbcUCC \Rightarrow aaabbbccUC \Rightarrow aaabbbccc \end{aligned}$$

Nākošais šādas sarežģītības valodas piemērs:

Valoda #9 $\{a^n \mid n \text{ ir nenegatīva divnieku pakāpe}\}$.

Katrs vārds sastāv no burtiem a, turklāt a skaits ir nenegatīva divnieku pakāpe: 0, 1, 2, 4, 8 utt.

Valodas #9 vārdu piemēri:

$a \quad aaaa \quad aaaaaaaaaaaaaaaaaa$

Att. 2-15. Gramatika Γ_9 (valodai $\{a^n \mid n \text{ ir nenegatīva divnieku pakāpe}\}$).

$\Gamma_9 = (\{a\}, \{S, Q, N, R\}, P_9, S)$

P_9 :

1. $S \rightarrow QNQ$
2. $QN \rightarrow QR$
3. $RN \rightarrow NNR$
4. $RQ \rightarrow NNQ$
5. $N \rightarrow a$
6. $Q \rightarrow \varepsilon$

Piemēram, vārdu

aaaa

ar gramatiku Γ_9 var ģenerēt šādā secībā:

$$S \xRightarrow{1.} QNQ \xRightarrow{2.} QRQ \xRightarrow{4.} QNNQ \xRightarrow{2.} QRNQ \xRightarrow{3.} QNNRQ \xRightarrow{4.} QNNNNQ \xRightarrow{5.} QaaaaQ \xRightarrow{6.} aaaa$$

2.3. Gramatiku klases

Atkarībā no sarežģītības gramatikas iedalās klasēs. Ir 0., 1., 2. un 3. tipa gramatikas (sākot no sarežģītākajām uz vienkāršākajām). Vienkāršākās gramatikas klases ietilpst sarežģītākajās (pēc līdzības ar to, kā ābols ir arī auglis), tādējādi var teikt, ka visas gramatikas pieder pie 0. tipa. Nākošā tabula parāda gramatiku klasifikāciju, sākot no sarežģītākas, ietverot arī sinonīmus klašu nosaukumiem un ierobežojumus.

Tab. 2-2. Gramatiku klasifikācija.

Gramatiku tips	Sinonīms	Ierobežojumi
0.	–	–
1.	konteksta atkarīgas	produkciju kreisās puses nedrīkst būt garākas par labajām pusēm (šīs divas produkcijas neatbilst: $QN \rightarrow R$, $S \rightarrow \epsilon$)
2.	bezkonteksta	tas pats + visām produkcijām kreisajā pusē drīkst būt tikai viens neterminālais simbols (ši produkcija neatbilst: $QN \rightarrow QR$)
3.	regulāras	tas pats + nesatur vidējo rekursiju (vidējā rekursija: $S \rightarrow aSa$)

2.4. Vingrinājumi

Vingrinājums #1.

Izveidot regulāru izteiksmi valodai, kas sastāv no vārdiem, kas sākas ar vienu burtu a, pēc kā seko viens vai vairāki burti b, bet beigās var būt (neobligāti) viens vai divi burti c. Šīs valodas vārdu piemēri: ab, abbbb, abcc, abbc.

Vingrinājums #2.

Uzrakstīt gramatiku valodai, kas aprakstīta iepriekšējā vingrinājumā.

Vingrinājums #3.

Uzrakstīt gramatiku valodai $\{W \mid W\text{-loģiska izteiksme, kas var sastāvēt no mainīgajiem } a, b, c \text{ un loģiskajām operācijām } \&, |, \text{ kā arī iekavām}\}$. Šīs valodas vārdu piemēri: c, b|c, a&b, a|(b&c)|b, c&(b&(a|c)).

Vingrinājums #4.

Uzrakstīt gramatiku valodai $\{(a|b)^* \mid \text{count}(a) = \text{count}(b)\}$. (Pie valodas pieder tādi vārdi, kas sastāv no burtiem a un b jebkurā secībā, turklāt a un b skaits ir vienāds, piemēram, abab, baaabb.)

Vingrinājums #5.

Izvest vārdu abbbbabaaa gramatikā, kas izveidota vingrinājumā #4.