Konstekstno slobodne gramatike

Vežbe 07 - PPJ

Nemanja Mićović
nemanja_micovic@matf.bg.ac.rs

Matematički fakultet, Univerzitet u Beogradu

4. decembar 2017

Sadržaj

Sadržaj

Sintaksna analiza

- Omogućava nam da proverimo sintaksu/pravopis jezika
- Želimo formalizam za opis sintakse koji je dovoljno izražajan [Aho et al., 1986]

Primer sintaksne analize

- gcc vrši sintaksnu analizu nad našim kodom
- Prijavljuje sintaksnu grešku

```
#include <stdio.h>

int main()
{
    int i = 0 j = 0;
    return 0;
}
```

```
test.c: In function 'main':
test.c:5:15: error: expected ',' or ';' before 'j'
int i = 0 j = 0;
          ^
```

Gramatike

- Formalizam za opis sintakse jezika
- Izražajniji mehanizam od regularnih izraza

Primer gramatike

- Pravila:
  - 1. $A \rightarrow aAb$
  - 2. $A \rightarrow \varepsilon$
- Primer izvođenja: $A \Rightarrow^1 aAb \Rightarrow^1 aaAbb \Rightarrow^1 aaaAbbb \Rightarrow^2 aaa\varepsilon bbb$
- Pravila definišu jezik: $L = \{a^n b^n | n \geq 0\}$[1]

[1] Primetimo da navedeni jezik nije moguće opisati regularnim izrazima jer nije regularan, dokaz sa predavanja.

Simboli u konstekstno slobodnoj[2] gramatici

Simboli mogu biti

- završni (terminalni simboli, terminali, tokeni)
- nezavršni (neterminalni simboli, neterminali, pomoćni simboli)

U prethodnom primeru:

- Završni: $\{a, b\}$
- Nezavršni: $\{A\}$

[2] Termin će biti objašnjen kasnije

Zadatak 1

- Opisati KS gramatikom jezik $L = \{a^n b^n | n > 0\}$

Zadatak 1

- Opisati KS gramatikom jezik $L = \{a^n b^n | n > 0\}$
- Rešenje

Zadatak 1

- Opisati KS gramatikom jezik $L = \{a^n b^n | n > 0\}$
- Rešenje
  - $A \rightarrow aAb$

Zadatak 1

- Opisati KS gramatikom jezik $L = \{a^n b^n | n > 0\}$
- Rešenje
  - $A \rightarrow aAb$
  - $A \rightarrow ab$

Zadatak 2

- Zapisati jezik `a*b*` koristeći KS gramatike
  - $S \rightarrow AB$
  - $A \rightarrow aA$
  - $A \rightarrow \varepsilon$
  - $B \rightarrow bB$
  - $B \rightarrow \varepsilon$

Definicija KS gramatike

KS Gramatika G je uređena četvorka (Σ, N, S, P), za koju važi

- Σ je skup završnih simbola
- N je skup nezavršnih simbola
- S je aksioma (početni simbol), $S \in N$
- P je skup pravila gramatike, $P \subseteq N \times (N \cup \Sigma)$

Definicija KS gramatike - primer

Gramatika:

- $A \to aAb$
- $A \to \varepsilon$

Skupovi:

- $\Sigma = \{a, b\}$
- $N = \{A\}$
- $S = A$
- $P = \{(A, aAb), (A, ab)\}$

Šta znači termin kontekstno oslobodna?

- Sa leve strane pravila, javlja se tačno **jedan neterminalni simbol**
- Inače, primena pravila zavisi od konteksta

Konstekstno slobodna gramatika:

- $A \to aAb$
- $A \to \varepsilon$

Gramatika koja nije kontekstno slobodna:

- $A \to aBcBb$
- $aB \to aaB$
- $Bb \to Bbb$

Rečenična forma

Rečenična forma je bilo koja rečenica iz $(N \cup \Sigma)*$:

- S je rečenična forma
- Ako je $\alpha X \beta$ rečenična forma i $X \rightarrow \gamma \in P$, tada je $\alpha\gamma\beta$ takođe rečenična forma

Tada pišemo $\alpha X \beta \Rightarrow \alpha\gamma\beta$ i kažemo da su ove dve rečenične forme u **relaciji izvođenja**.

Relacije izvođenja

- $\to$ koristimo za **pravila gramatike**
- $\Rightarrow$ koristimo za relaciju izvođenja rečeničnih formi
- $\Rightarrow^+$ tranzitivno zatvorenje (izvodi se u jednom ili više koraka)
- $\Rightarrow^*$ tranzitivno i refleksivno zatvorenje (izvodi se u nula ili više koraka)

Jezik gramatike

Jezik gramatike G možemo definisati na sledeći način:

$$L(G) = \{w \in \Sigma^* | S \Rightarrow^* w\}$$

Izvođenje u gramatici

Podsetimo se gramatike G jezika $L = \{a^n b^n | n \geq N\}$

- Pravila gramatike:
  - 1. $A \rightarrow aAb$
  - 2. $A \rightarrow \varepsilon$
- Primer izvođenja: $A \Rightarrow^1 aAb \Rightarrow^1 aaAbb \Rightarrow^1 aaaAbbb \Rightarrow^2 aaa\varepsilon bbb \Rightarrow aaabbb$
- Što kraće možemo zapisati kao $A \Rightarrow^* aaabbb \in L(G)$

Izvođenje u gramatici

- **Izvođenje u gramatici G** je niz rečeničnih formi R_i gde je **prva aksioma (S)**, **poslednja reč iz jezika (w)**, i svake dve su u **relaciji izvođenja** ($\Rightarrow$).
- $S \Rightarrow R_1 \Rightarrow R_2 \Rightarrow ... \Rightarrow R_n \Rightarrow w$
- Programski prevodioci pri sintaksnoj analizi izvođenjem koristeći pravila gramatike proveravaju sintaksu programskog koda

Sadržaj

Aritmetički izrazi

- Prepoznajemo aritmetičke izraze
- `1 + 9 + 10`
- `100 + 200 + 300`
- Radi apstrakcije, brojeve ćemo označavati sa a

Aritmetički izrazi - rešenje 1

Uočimo sledeću gramatiku[3] G_{+R}:

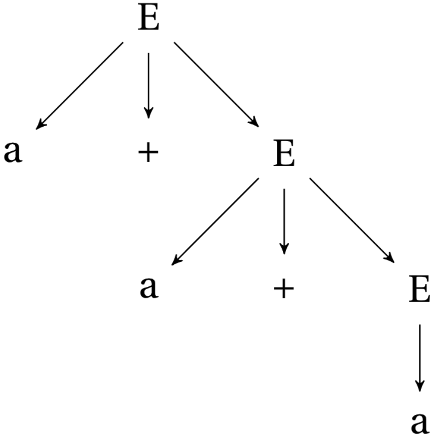
- 1. $E \rightarrow a$
- 2. $E \rightarrow a + E$

Izvedimo izraz $a + a + a$:

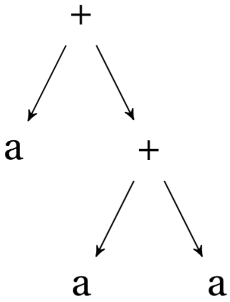
$$E \Rightarrow^2 a + E \Rightarrow^2 a + a + E \Rightarrow^1 a + a + a$$

[3] E potiče od engleskog naziva *expression* za *izraz*

Aritmetički izrazi - rešenje 1



Slika: Drvo izvođenja



Slika: Drvo apstraktne sintakse

Izrazi - rešenje 1

- Gramatika G_{+R} je **desno rekurzivna**
- Primetimo da su prikazana drveta *nagnuta* na desno

Aritmetički izrazi - rešenje 2

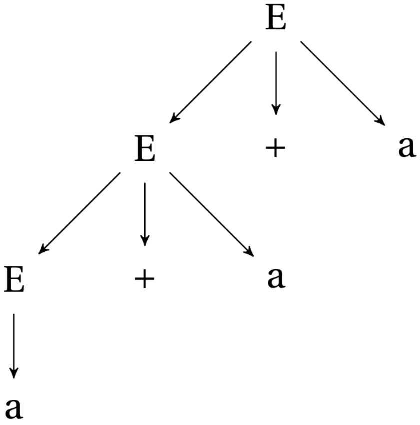
Uočimo sledeću gramatiku G_{+L}:

- 1. $E \rightarrow a$
- 2. $E \rightarrow E + a$

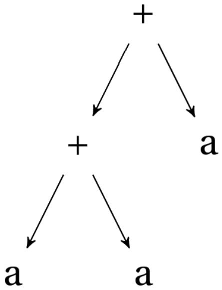
Izvedimo izraz $a + a + a$:

$$E \Rightarrow^2 E + a \Rightarrow^2 E + a + a \Rightarrow^1 a + a + a$$

Aritmetički izrazi - rešenje 2



Slika: Drvo izvođenja



Slika: Drvo apstraktne sintakse

Aritmetički izrazi - rešenje 2

- Gramatika G_{+L} je **levo rekurzivna**
- Primetimo da su prikazana drveta *nagnuta* na levo

Aritmetički izrazi - rešenje 3

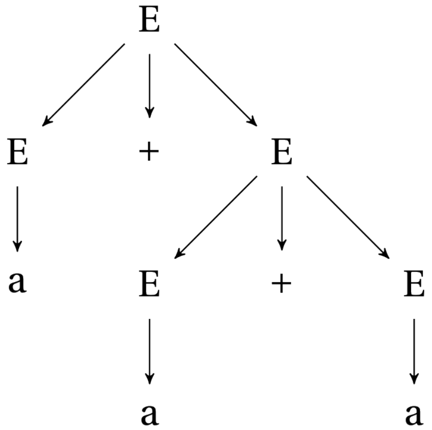
Uočimo sledeću gramatiku G_{+LD}:

- 1. $E \to a$
- 2. $E \to E + E$

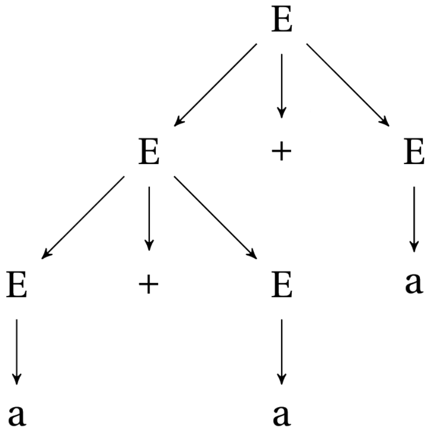
Izvedimo izraz $a + a + a$:

- Najlevlje izvođenje:
  $E \Rightarrow^2 E + E \Rightarrow^1 a + E \Rightarrow^2 a + E + E \Rightarrow^1 a + a + E \Rightarrow^1 a + a + a$
- Najdešnje izvođenje:
  $E \Rightarrow^2 E + E \Rightarrow^1 E + a \Rightarrow^2 E + E + a \Rightarrow^1 a + a + a \Rightarrow^1 a + a + a$

Aritmetički izrazi - rešenje 3

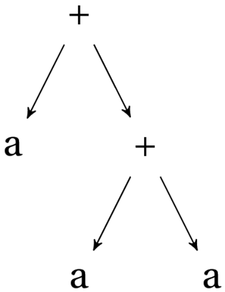


Slika: Drvo izvođenja za najlevlje izvođenje

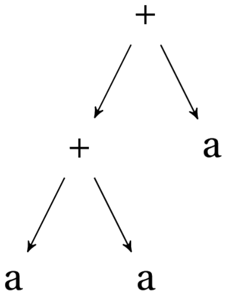


Slika: Drvo izvođenja za najdešnje izvođenje

Aritmetički izrazi - rešenje 3



Slika: Drvo apstraktne sintakse za najlevlje izvođenje



Slika: Drvo apstraktne sintakse za najdešnje izvođenje

Aritmetički izrazi - rešenje 3

- Gramatika G_{+LD} ima dva različita drveta izvođenja za isti skup pravila!
- Takvu gramatiku nazivamo **višeznačna gramatika**
- Ispitivanje da li je gramatika **višeznačna** je **neodlučiv** problem
- Ipak, ukoliko postoji pravilo koje je i levo i desno rekurzivno, to je odličan indikator o višeznačnosti

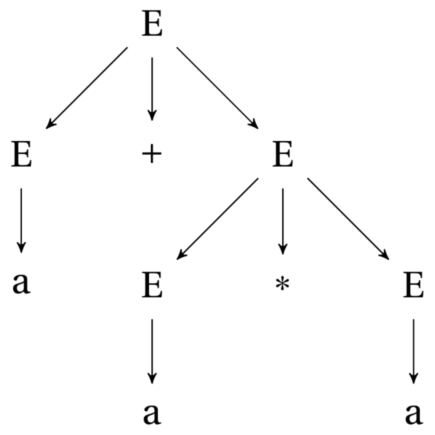
Aritmetički izrazi 2

- Želimo da omogućimo i množenje
- Analizirajmo sledeću gramatiku G_{+*LD}:
  1. $E \to a$
  2. $E \to E + E$
  3. $E \to E * E$
- Ili kraće: $E \to a \mid E + E \mid E * E$

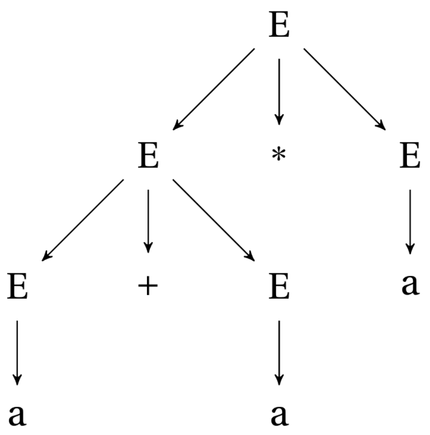
Aritmetički izrazi 2

- Izvedimo $a + a * a$
- Ako prvo koristimo drugo pravilo:
  $I_1 : E \Rightarrow^2 E + E \Rightarrow^1 a + E \Rightarrow^3 a + E * E \Rightarrow^1 a + a * E \Rightarrow^1 a + a * a$
- Ako prvo koristimo treće pravilo:
  $I_2 : E \Rightarrow^3 E * E \Rightarrow^2 E + E * E \Rightarrow^1 a + E * E \Rightarrow^1 a + a * E \Rightarrow^1 a + a * a$

Aritmetički izrazi 2 - drveta



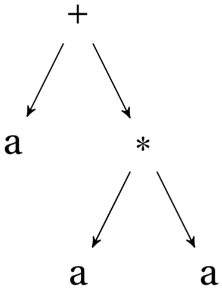
Slika: Drvo izvođenja za I_1



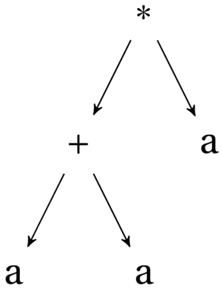
Slika: Drvo izvođenja za I_2

Aritmetički izrazi 2 - drveta

- Koje od prikazanih drveta je ispravno ako se uzme u obzir da množenje ima veći prioriet od sabiranja?



Slika: Drvo apstraktne sintakse za I_1



Slika: Drvo apstraktne sintakse za I_2

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \rightarrow E + T$
  2. $E \rightarrow T$

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$
  2. $E \to T$
  3. $T \to T * a$

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$
  2. $E \to T$
  3. $T \to T * a$
  4. $T \to a$

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$
  2. $E \to T$
  3. $T \to T * a$
  4. $T \to a$
- Kraće: $E \to E + T \mid T$ i $T \to T * a \mid a$

Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
    1. $E \to E + T$
    2. $E \to T$
    3. $T \to T * a$
    4. $T \to a$
- Kraće: $E \to E + T \mid T$ i $T \to T * a \mid a$
- Gubimo **višeznačnost**

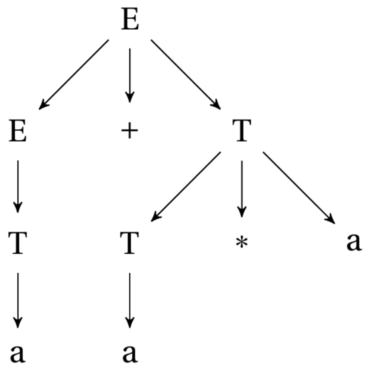
Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$
  2. $E \to T$
  3. $T \to T * a$
  4. $T \to a$
- Kraće: $E \to E + T \mid T$ i $T \to T * a \mid a$
- Gubimo **višeznačnost**
- Rešavamo problem prioriteta operacija

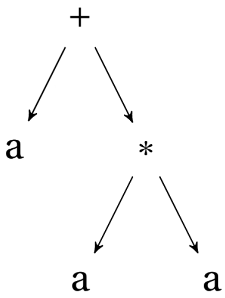
Aritmetički izrazi 2 - pokušaj 2

- Izraze posmatramo kao niz sabiraka
- Svaki sabirak je ili broj ili proizvod brojeva
- To nas inspiriše za gramatiku G_{+*L}
  1. $E \to E + T$
  2. $E \to T$
  3. $T \to T * a$
  4. $T \to a$
- Kraće: $E \to E + T \mid T$ i $T \to T * a \mid a$
- Gubimo **višeznačnost**
- Rešavamo problem prioriteta operacija
- $I : E \Rightarrow^1 E + T \Rightarrow^2 T + T \Rightarrow^4 a + T \Rightarrow^3 a + T * a \Rightarrow^4 a + a * a$

Aritmetički izrazi 2 - pokušaj 2



Slika: Drvo apstraktne sintakse za I

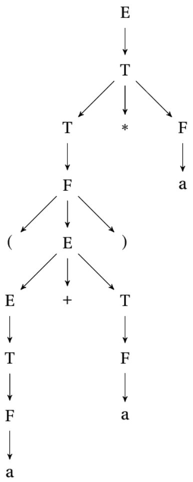


Slika: Drvo apstraktne sintakse za I

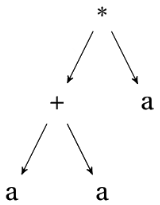
Aritmetički izrazi 3 - kompletirani

- Želimo da dodamo −, / i zagrade
- Gramatika G_{opz}
  - $E \rightarrow E + T \mid E - T \mid T$
  - $T \rightarrow T * F \mid T/F \mid F$
  - $F \rightarrow a \mid (E)$
- Izvedimo $(a + a) * a$
  - $E \Rightarrow T \Rightarrow T * F \Rightarrow F * F \Rightarrow (E) * F \Rightarrow (E + T) * F \Rightarrow (T + T) * F$
  - $(T + T) * F \Rightarrow (F + T) * F \Rightarrow (a + T) * F \Rightarrow (a + F) * F \Rightarrow (a + a) * F \Rightarrow (a + a) * a$

Aritmetički izrazi 3 - kompletnirani



Slika: Drvo apstraktne sintakse za *I*



Slika: Drvo apstraktne sintakse za *I*

Lista brojeva

- Konstekstno slobodnom gramatikom opisati[4] liste brojeva.
- Na primer[5], `1, 8, 5, 6, 1, 9, 4, 3`
  - $BROJ \rightarrow 1|2|3|4|5|6|7|8|9$

[4] Koristićemo simbol | kako bi izbegli suvišna ponavljanja pravila koje imaju istu levu stranu.
[5] Vidite li nešto interesantno u brojevima?

Lista brojeva

- Konstekstno slobodnom gramatikom opisati[4] liste brojeva.
- Na primer[5], `1, 8, 5, 6, 1, 9, 4, 3`
  - $BROJ \rightarrow 1|2|3|4|5|6|7|8|9$
  - $lista \rightarrow BROJ, lista$

[4] Koristićemo simbol `|` kako bi izbegli suvišna ponavljanja pravila koje imaju istu levu stranu.
[5] Vidite li nešto interesantno u brojevima?

Lista brojeva

- Konstekstno slobodnom gramatikom opisati[4] liste brojeva.
- Na primer[5], `1, 8, 5, 6, 1, 9, 4, 3`
  - $BROJ \to 1|2|3|4|5|6|7|8|9$
  - $lista \to BROJ, lista$
  - $lista \to BROJ$

[4]Koristićemo simbol `|` kako bi izbegli suvišna ponavljanja pravila koje imaju istu levu stranu.
[5]Vidite li nešto interesantno u brojevima?

Lista brojeva

- Konstekstno slobodnom gramatikom opisati[4] liste brojeva.
- Na primer[5], `1, 8, 5, 6, 1, 9, 4, 3`
  - $BROJ \rightarrow 1|2|3|4|5|6|7|8|9$
  - $lista \rightarrow BROJ, lista$
  - $lista \rightarrow BROJ$
  - Ili kraće: $lista \rightarrow BROJ, lista|BROJ$

[4] Koristićemo simbol `|` kako bi izbegli suvišna ponavljanja pravila koje imaju istu levu stranu.
[5] Vidite li nešto interesantno u brojevima?

Sadržaj

Čišćenje suvišnih simbola - nedostižni simboli

- Nedostižni simboli su svi simboli koji nisu dostižni
- Pronalaženjem dostižnih simbola saznajemo koji su nedostižni
  - Nivo 1: Aksioma je dostižan simbol
  - Nivo 2: Svi simboli koje se nalaze sa desne strane aksiome su dostižni
  - ...
  - Nivo n: Svi simboli koje se nalaze sa desne strane pravila iz prethodnoh nivoa su dostižna

Čišćenje suvišnih simbola - nedostižni simboli

- $A \rightarrow aB \mid bA$
- $B \rightarrow cD \mid E$
- $C \rightarrow CA \mid Ba$
- $D \rightarrow a \mid b$
- $E \rightarrow eE \mid Ee$
- $F \rightarrow Ca \mid Fb$

Dostižni simboli:

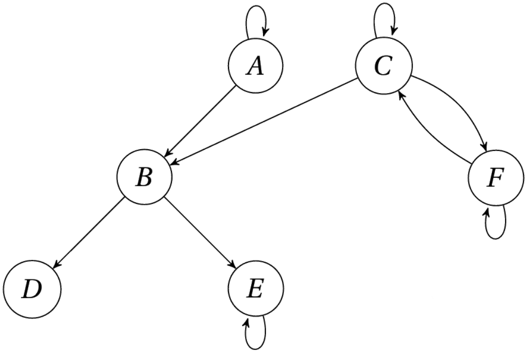
1. nivo: A
2. nivo: A, B
3. nivo: A, B, D, E
4. nivo: A, B, D, E

Nedostižni:

- C, F (jer nisu dostižni)

Čišćenje suvišnih simbola - nedostižni simboli

- $A \to aB \mid bA$
- $B \to cD \mid E$
- $C \to CA \mid Ba$
- $D \to a \mid b$
- $E \to eE \mid Ee$
- $F \to Ca \mid Fb$



Slika: Dostižni simboli

Čišćenje suvišnih simbola - neproduktivni simboli

- Neproduktivni simboli su oni koji imaju rekurzivno pravilo a **nemaju** izlaz iz rekurzije
- Neproduktivne simbole ćemo naći tako što pronađemo prvo produktivne
  - Nivo 1: Simboli koji ne izvode neterminale su produktivni simboli
  - ...
  - Nivo n: Simboli koji izvode terminale i produktivne simbole iz prethodnog nivoa.

Čišćenje suvišnih simbola - neproduktivni simboli

- $A \rightarrow aB \mid bA$
- $B \rightarrow cD \mid E$
- $D \rightarrow a \mid b$
- $E \rightarrow eE \mid Ee$

Produktivni simboli:

1. nivo: D
2. nivo: D, B
3. nivo: D, B, A
4. nivo: D, B, A

Neproduktivni:

- E (jer nije produktivan)

Čišćenje suvišnih simbola - konačna gramatika

Polazna gramatika:

- $A \to aB \mid bA$
- $B \to cD \mid E$
- $C \to CA \mid Ba$
- $D \to a \mid b$
- $E \to eE \mid Ee$
- $F \to Ca \mid Fb$

Redukovana gramatika:

- $A \to aB \mid bA$
- $B \to cD \mid E$
- $D \to a \mid b$

Oslobađanje jednostrukih pravila

- Pravilo je jednostruko ako ima samo jedno izvođenje (samo jednu desnu stranu)
- Oslobađamo se tako što desnu stranu jedostrukog pravila uvrstimo umesto leve strane pravila u svim ostalim pravilima

Polazna gramatika:

- $A \to aB \mid bA$
- $B \to cD \mid E$
- $C \to CA \mid Ba$
- $D \to a \mid b$
- $E \to eE \mid Ee$
- $F \to Ca \mid Fb$

Redukovana gramatika:

- $A \to aB \mid bA$
- $B \to cD$
- $D \to a \mid b$

Nakon oslobađanja jednostrukih pravila:

- $A \to acD \mid bA$
- $D \to a \mid b$

Oslobađanje jednostrukih pravila

- Gramatiku možemo još dodatno skratiti (ovaj put više iz zabave)
- Koji jezik opisuje gramatika? (Napisati regularni izraz)

Polazna gramatika:

- $A \to aB \mid bA$
- $B \to cD \mid E$
- $C \to CA \mid Ba$
- $D \to a \mid b$
- $E \to eE \mid Ee$
- $F \to Ca \mid Fb$

Izuzetno redukovana gramatika:

- $A \to aca \mid acb \mid bA$

Oslobađanje leve rekurzije

- Pravilo je **levo-rekurzivno** ako je neterminal sa leve strane **prvi simbol** sa desne strane pravila
- Jezik $\beta\alpha*$ opisan je sledećom gramatikom: $A \to A\alpha \mid \beta$
- Leve rekurzije se oslobađamo uvođenjem novog neterminala:
  - $A \to \beta A'$
  - $A' \to \alpha A' \mid \varepsilon$
- Eliminišemo epsilon:
  - $A \to \beta A' \mid \beta$
  - $A' \to \alpha A' \mid \alpha$

Oslobađanje leve rekurzije - opšti postupak

- Jezik $(\beta_1|\beta_2|...|\beta_m)(\alpha_1|\alpha_2|...|\alpha_n)*$
- Gramatika: $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid ... \mid A\alpha_n \mid \beta_1 \mid \beta_2 \mid ... \mid \beta_m$
- Nakon oslobađanja:
  - $A \rightarrow \beta_1 A' \mid \beta_2 A' \mid ... \mid \beta_m A'$
  - $A' \rightarrow \alpha_1 A' \mid \alpha_2 A' \mid ... \mid \alpha_n A' \mid \varepsilon$
- Nakon eliminisanja ε pravila:
  - $A \rightarrow \beta_1 A' \mid \beta_2 A' \mid ... \mid \beta_m A' \mid \beta_1 \mid \beta_2 \mid ... \mid \beta_m$
  - $A' \rightarrow \alpha_1 A' \mid \alpha_2 A' \mid ... \mid \alpha_n A' \mid \alpha_1 \mid \alpha_2 \mid ... \mid \alpha_n$

Oslobađanje od ε pravila

Gramatika je ε **slobodna**

- ako nema ε pravilo ili ako se ono javlja samo u aksiomi, pri čemu aksioma ne sme da se javlja sa desne strane pravila

Anulirajući simboli su oni simboli koji mogu da izvedu ε u nula ili više koraka ($A \Rightarrow^* \varepsilon$)

Oslobađanje od ε pravila

Polazna gramatika:

- $S \rightarrow Aa \mid Bba$
- $A \rightarrow aBB \mid CC \mid aD$
- $B \rightarrow aB \mid b$
- $C \rightarrow CD \mid DE \mid a$
- $D \rightarrow aB \mid bBa \mid \varepsilon$
- $E \rightarrow aD \mid DD$

Tražimo simbole koji izvode ε, a potom i simbole koji izvode pronađene anulirajuće simbole

1. $A_0 = \{D\}$
2. $A_1 = \{D, E\}$
3. $A_2 = \{D, E, C\}$
4. $A_3 = \{D, E, C, A\}$
5. $A_4 = \{D, E, C, A\}$

Oslobađanje od ε pravila

Svaka desna strana se piše više puta uzimajući u obzir da li anulirajući simbol može da se javi ili ne:

- pravilo S: A je anulirajući simbol pa možemo izvesti a (ako A izvede ε) ili Aa (ako A ne izvede ε)
- pravilo A:
  - –C je anulirajući simbol pa možemo izvesti C ili CC
  - –D je anulirajući simbol pa možemo izvesti a ili aD
- pravilo B: nema anulirajućih simbola pa ga prepisujemo
- pravilo C:
  - C i D su anulirajući pa možemo izvesti D, C, CD i εC i D su anulirajući pa možemo izvesti D, C, CD i ε (nećemo pisati ε)
  - D i E su anulirajući pa možemo izvesti D, E ili DE
- pravilo E: D je anulirajući pa možemo da izvedemo a, aD, D ili DD

Oslobađanje od ε pravila

Dobija se sledeća gramatika:

- $S \to a \mid Aa \mid Bba$
- $A \to aBB \mid C \mid CC \mid a \mid aD$
- $B \to ab \mid b$
- $C \to D \mid CD \mid E \mid DE \mid a$
- $D \to aB \mid bBa$
- $E \to a \mid aD \mid D \mid DD$

Literatura I

Aho, A. V., Sethi, R., and Ullman, J. D. (1986).
Compilers principles, techniques, and tools.
Addison-Wesley, Reading, MA.