Formalne metode

u softverskom inženjerstvu

11 potisni automati

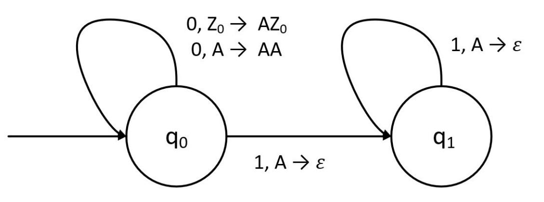
ETFBL 24-25

Dunja Vrbaški

PDA za 0^n1^n

- dva stanja (za nule i jedinice)
- svaka pročitana 0
    - automat: isto stanje (inicijalno)
    - glava: pomera se
    - stek: odgovarajući simbol (00 umesto 0)
- svaka pročitana 1
    - automat: isto stanje (stanje za 1)
    - glava: pomera se
    - stek: pop
- prelaz sa 0 na 1

$q_0 0\$ \to q_0 R\$S$	push S onto the stack
$q_0 0S \to q_0 RSS$	push S onto the stack
$q_0 1\$ \to q_0 N\$$	first symbol in the input is 1; loop forever
$q_0 1S \to q_1 R\epsilon$	first 1 is encountered
$q_0 \square\$ \to q_0 N\epsilon$	input string is empty; accept
$q_0 \square S \to q_0 NS$	input only consists of 0s; loop forever
$q_1 0\$ \to q_1 N\$$	0 to the right of 1; loop forever
$q_1 0S \to q_1 NS$	0 to the right of 1; loop forever
$q_1 1\$ \to q_1 N\$$	too many 1s; loop forever
$q_1 1S \to q_1 R\epsilon$	pop top symbol from the stack
$q_1 \square\$ \to q_1 N\epsilon$	accept
$q_1 \square S \to q_1 NS$	too many 0s; loop forever



prihvatanje praznim stekom

Rekli smo:

- CF jezici se prepoznaju PDA automatima

Tvrđenje:
Jezik je kontekstno nezavisan akko postoji PDA koji prihvata taj jezik.

Treba pokazati:

- za proizvoljnu CFG postoji odgovarajući PDA
- za proizvoljni PDA postoji odgovarajuća CFG

CFG $\rightarrow$ PDA
Ideja: parsiranje

Posebne forme CFG gramatika

- dokazi
- automatizacija parsiranja

Pojednostavljenje gramatike:

- odbacivanje beskorisnih simbola
- odbacivanje e-produkcija
- odbacivanje jediničnih produkcija

$G = (\Sigma, N, P, S)$

Simbol $X \in (N \cup \Sigma)$ je koristan ako se koristi u postupku generisanja reči w: $S \rightarrow^* \alpha X \beta \rightarrow^* w$

X može biti beskoristan ako je:

- **negenerativan**

  ako iz tog simbola nije moguće generisati neku reč $w \in \Sigma^*$,
  tj. ne postoji postupak generisanja $X \rightarrow^* w$

- **nedohvatljiv**

  ako se taj simbol ne nalazi ni u jednom nizu koji se generiše iz početnog neterminalnog simbola, tj. ako ne postoji postupak generisanja $S \rightarrow^* \alpha X \beta$

Odbacivanje negenerativnih simbola

1. U listu generativnih simbola se stavljaju leve strane produkcija koje na desnoj strani imaju samo terminalne simbole.

2. U listu se dodaju neterminalni simboli sa leve strane produkcija koje na desnoj strani imaju isključivo generativne simbole.

3. Algoritam se zaustavlja ako nije moguće proširiti listu generativnih simbola. Svi neterminalni simboli koji se ne nalaze u listi generativnih simbola su negenerativni simboli.

Data je gramatika $G=(N,\Sigma,P,S)$ sa sledećim produkcijama:

$S \rightarrow Aa|aBb$
$A \rightarrow bAb$
$B \rightarrow bBb|A|a$
$C \rightarrow a|c$

- u prvom koraku, simboli B i C se stavljaju u listu generativnih simbola
- dalje, na osnovu drugog koraka, se u listu dodaje simbol S
  (jer produkcija $S \rightarrow aBb$ sadrži samo generativne simbole)
- lista se više ne može proširivati, pa simbol A nije generativan simbol

Ekvivalentna gramatika $G_1=(N_1, \Sigma_1, P_1, S)$ ima sledeće produkcije:

$S \rightarrow aBb$
$B \rightarrow bBb|a$
$C \rightarrow a|c$

Odbacivanje nedohvatljivih simbola

1. Početni neterminalni simbol se stavlja u listu dohvatljivih simbola.

2. U listu se dodaju svi simboli sa desne strane produkcija koje na levoj strani imaju simbol koji je u listi dohvatljivih simbola.

3. Algoritam se zaustavlja ako nije moguće proširiti listu dohvatljivih simbola. Svi simboli koji nisu u listi su nedohvatljivi.

Data je gramatika $G_1=(N_1, \Sigma_1, P_1, S)$, iz prethodnog primera, sa produkcijama:

$S \to aBb$
$B \to bBb|a$
$C \to a|c$

- u prvom koraku, simbol S se stavlja u listu dohvatljivih simbola
- dalje, na osnovu drugog koraka, se u listu dodaju simboli a, B i b (na osnovu prve produkcije)
- lista se proširuje simbolima sa desne strane druge produkcije (jer je simbol B, koji je već u listi, na levoj strani)
- svi simboli su već u listi, pa se algoritam završava

Ekvivalentna gramatika $G_2=(N_2, \Sigma_2, P_2, S)$ ima sedeće produkcije:

$S \to aBb$
$B \to bBb|a$

Odbacivanje svih beskorisnih simbola se postiže sledećim redosledom:

1) odbacivanje negerativnih simbola,
2) odbacivanje nedohvatljivih simbola

Može se pokazati da se zamenom redosleda koraka u odbacivanju beskorisnih simbola ne dobije isti rezultat za svaku gramatiku.

Data je gramatika $G=(N,\Sigma,P,S)$ sa produkcijama

$S \to AB \mid a$
$A \to a$

- odbacivanjem negenerativnih, a onda nedohvatljivih simbola, dobija se gramatika sa jednom produkcijom: $S \to a$

- odbacivanjem nedohvatljivih, pa onda negenerativnih simbola, dobija se gramatika sa dve produkcije: $S \to a$ i $A \to a$

Odbacivanje ε-produkcija

Neka je data gramatika *G=(N,Σ,P,S)* sa skupom produkcija

$S \rightarrow aAb$
$A \rightarrow aAb$
$A \rightarrow \varepsilon$

Treća produkcija predstavlja ε-produkciju.

Neterminalni simbol je **prazan simbol** ako postoji postupak generisanja $A \rightarrow^* \varepsilon$

Postupak pronalaženja praznih simbola:

1. listu praznih simbola se dodaju neterminalni simboli koji su sa leve strane ε-produkcija
2. u listu praznih simbola se dodaju neterminalni simboli sa leve strane produkcija koje sa desne strane imaju isključivo simbole koji su već u listi

Postupak odbacivanja ε-produkcija:

1. Pronalaze se svi prazni simboli.

2. Neterminalni simboli (N), terminalni simboli (Σ) i startni simbol (S) su isti kao i kod originalne gramatike

3. Produkcije nove gramatike se grade tako da se za svaku produkciju $A \to X_1 X_2 \ldots X_n$ gramatike G u gramatiku G' dodaje produkcija $A \to Y_1 Y_2 \ldots Y_n$ pri čemu simboli Y_i imaju sledeće vrednosti:
   a. Y_i dobija vrijednost X_i ako simbol X_i nije prazan simbol,
   b. Y_i dobija vrijednost ε ili X_i ako je simbol X_i prazan simbol

4. Produkcije sa simbolima Y_i koje su ε-produkcije se ne dodaju u skup novih produkcija

5. Kod uvođenja Y_i simbola, moraju se uzeti sve kombinacije praznih simbola iz skupa

6. Sve ε-produkcije se mogu odbaciti osim jedne; skup P' se mora proširiti produkcijom $S \to \varepsilon$, čime se u generisani jezik uključuje prazna reč ako je jezik sadrži

Data je gramatika *G=(N,Σ,P,S)* sa produkcijama:

$S \rightarrow ABCD|E$
$A \rightarrow a|\varepsilon$
$B \rightarrow b|c$
$C \rightarrow b|\varepsilon$
$D \rightarrow a|b$
$E \rightarrow a|b|\varepsilon$

- u skup praznih simbola se prvo dodaju simboli A, C i E, jer se nalaze sa leve strane ε-produkcija
- u sledećem koraku se skup proširuje simbolom S, jer postoji produkcija S→E (E je već u skupu praznih simbola).
- To je ujedno i konačan skup praznih simbola: {A, C, E, S}

$S \rightarrow ABCD | E$
$A \rightarrow a | \varepsilon$
$B \rightarrow b | c$
$C \rightarrow b | \varepsilon$
$D \rightarrow a | b$
$E \rightarrow a | b | \varepsilon$

skup praznih simbola: {A, C, E, S}

- posmatra se produkcija $S \rightarrow ABCD$.
  Sa desne strane se nalaze prazni simboli A i C, pa se dodaju sedeće kombinacije:

  $S \rightarrow \varepsilon B \varepsilon D | \varepsilon BCD | AB\varepsilon D | ABCD$ odnosno
  $S \rightarrow BD | BCD | ABD | ABCD$

- iz produkcije $S \rightarrow E$ se zamenom dobija

  $S \rightarrow E | \varepsilon$, odnosno
  $S \rightarrow E$ (jer se ε produkcije ne dodaju u novi skup)

- ostale produkcije nemaju praznih simbola sa desne strane, pa se dodaju u novi skup produkcija (izuzimajući $A \rightarrow \varepsilon$, $C \rightarrow \varepsilon$ i $E \rightarrow \varepsilon$)

- pošto je u jeziku L(G) i prazna reč, dodaje se produkcija $S \rightarrow \varepsilon$

Odbacivanje jediničnih produkcija

Produkcije oblika **A→B**, gdje su A i B neterminalni simboli nazivaju se **jedinične produkcije**.

Jedinične produkcije komplikuju generisanje riječi jer unose dodatni (nepotreban) korak.
Postojanje jedinične produkcije ne mora značiti da je simbol koji je deo te produkcije beskoristan.

Npr. neka su date produkcije:

$A \rightarrow B$
$B \rightarrow w|C$

Ekvivalentan skup produkcija je:

$A \rightarrow w|C$

Novi skup produkcija stvara još jednu jediničnu produkciju (A→C), koja se u sedećoj iteraciji takođe može odbaciti.

Unit(A) = $\{B \in N \mid A \to^{*} B$ primenom samo jediničnih produkcija$\}$

Članovi skupa Unit(A) se izračunavaju na sedeći način:

1. u listu se doda neterminalni simbol A za koji se izračunava sadržaj skupa
2. lista se proširuje neterminalnim simbolima Y iz skupa N, takvi da postoji produkcija $X \to Y$, a X se već nalazi u listi
3. završava se kad nema više proširenja

Postupak odbacivanja jediničnih produkcija:

1. Skup novih produkcija P' se inicijalizuje produkcijama iz starog skupa P.
2. Izračunavaju se skupovi Unit(A) za svako A $\in$ N.
3. Za svaki element B iz skupa Unit(A) za koji postoji produkcija B→w, u skup P' se dodaje produkcija A→w.
4. Iz skupa P' se izbacuju sve jedinične produkcije.

Data je gramatika *G=(N,Σ,P,S)* sa skupom produkcija:

$S \rightarrow A|Aa$
$A \rightarrow B$
$B \rightarrow C|b$
$C \rightarrow D|ab$
$D \rightarrow b$

Skupovi jediničnih produkcija za sve neterminalne simbole:

Unit(S) = {S, A, B, C, D}
Unit(A) = {A, B, C, D}
Unit(B) = {B, C, D}
Unit(C) = {C, D}
Unit(D) = {D}

$S \rightarrow A \mid Aa$	Unit(S) = {S, A, B, C, D}
$A \rightarrow B$	Unit(A) = {A, B, C, D}
$B \rightarrow C \mid b$	Unit(B) = {B, C, D}
$C \rightarrow D \mid ab$	Unit(C) = {C, D}
$D \rightarrow b$	Unit(D) = {D}

Osnovni skup produkcija se proširuje produkcijama:

$S \rightarrow b \mid ab$

$A \rightarrow b \mid ab$

$B \rightarrow ab$

$C \rightarrow b$

Nakon odbacivanja jediničnih produkcija, dobija se:

$S \rightarrow b \mid ab \mid Aa$

$A \rightarrow b \mid ab$

$B \rightarrow b \mid ab$

$C \rightarrow b \mid ab$

$D \rightarrow b$

Iako nije traženo, gramatiku je moguće i dodatno pojednostaviti:

$S \rightarrow b \mid ab \mid Aa$

$A \rightarrow b \mid ab$

Normalna forma Čomskog

Gramatika G je u redukovanoj normalnoj formi Čomskog ako sve produkcije imaju sledeći oblik:

$A \rightarrow BC$
$A \rightarrow a$

$A, B, C \in N$, $a \in \Sigma$

Redukovana - ako gramatika ne sadrži ε-produkciju.
Gramatika u NFČ ne sme sadržavati beskorisne simbole.

Postupak transformacije u NFČ:

- U novi skup neterminalnih simbola N' se stavljaju svi neterminalni simboli iz skupa N.
- U skup produkcija P' se stavljaju sve produkcije koje su već u formi $A \to BC$ ili $A \to a$.
- Preostale produkcije se transformišu tako da se na desnoj strani nalaze isključivo terminalni ili isključivo neterminalni simboli.
    - Ako je produkcija A u obliku $A \to X_1 X_2 \ldots X_m$ gde je je $X_i = a_i$ terminalni simbol, u skup neterminalnih simbola se dodaje novi simbol C_a , a u skup produkcija $C_a \to a_i$ za svako X_i. Postupak se ponavlja sve dok sa desne strane ne ostanu samo neterminali.
    - Ako je podukcija A u obliku $A \to B_1 B_2 \ldots B_m$ gde je B_i neterminalni simbol, u skup neterminalnih simbola se dodaju novi simboli: $D_1, D_2, \ldots, D_{m-2}$, a zatim se produkcija A menja sjedećim produkcijama: $A \to B_1 D_1$, $D_1 \to B_2 D_2$, $D_2 \to B_3 D_3, \ldots,$ $D_{m-2} \to B_{m-1} B_m$

Data je gramatika *G=(N,Σ,P,S)* sa skupom produkcija:

$S \to ABa$
$A \to aab$
$B \to Ac$

Prvo se formira novi skup neterminalnih simbola N' u koji ulaze svi simboli iz N, kao i skup novih produkcija P' u koji ulaze sve produkcije iz P koje su već u formi koja odgovara NFČ (u ovom slučaju je to prazan skup).

Za svaki terminalni simbol se uvodi po jedan neterminalni simbol:

$N' = \{S, A, B\} \cup \{C_a, C_b, C_c\}$

$S \rightarrow ABa$
$A \rightarrow aab$
$B \rightarrow Ac$

$N' = \{S, A, B\} \cup \{C_a, C_b, C_c\}$

Skup produkcija P':

$S \rightarrow ABC_a$
$A \rightarrow C_aC_aC_b$
$B \rightarrow AC_c$
$C_a \rightarrow a$
$C_b \rightarrow b$
$C_c \rightarrow c$

$S \rightarrow ABa$
$A \rightarrow aab$
$B \rightarrow Ac$

$N' = \{S, A, B\} \cup \{C_a, C_b, C_c\}$

Skup produkcija P':

$S \rightarrow ABC_a$
$A \rightarrow C_aC_aC_b$
$B \rightarrow AC_c$
$C_a \rightarrow a$
$C_b \rightarrow b$
$C_c \rightarrow c$

Zbog prve i druge produkcije se uvode novi neterminalni simboli D_1 i D_2,

Konačan skup produkcija P’:

$S \rightarrow AD_1$
$D_1 \rightarrow BC_a$
$A \rightarrow C_aD_2$
$D_2 \rightarrow C_aC_b$
$B \rightarrow AC_c$
$C_a \rightarrow a$
$C_b \rightarrow b$
$C_c \rightarrow c$

Grajbahova normalna forma

Gramatika *G=(N,Σ,P,S)* je u Grajbahovoj normalnoj formi ako su sve produkcije u obliku:

$S \rightarrow a\alpha,\ a \in \Sigma,\ \alpha \in N$

Gramatika u Grajbahovoj normalnoj formi nema levih rekurzija.
Uklanja se potencijalni problem kod nekih parsera.

Za svaku kontekstno nezavisnu gramatiku G koja generiše jezik L(G)\{ε}, moguće je konstruisati gramatiku u Grajbahovoj normalnoj formi.

$L=\{a^n b^n \mid n \geq 1\}$

$S \rightarrow aSb$
$S \rightarrow ab$

$S \rightarrow aSB$
$S \rightarrow aB$
$B \rightarrow b$

CFL:

- svi regularni jezici
- balansirane zagrade
- palindromi
- većina poznate sintakse PJ
- ...

$a^n b^m$	RL	
$a^n b^n$	CFL	
$a^n b^n c^m$	CFL	
$a^n b^m c^m$	CFL	
$a^n b^n c^n$	CSL	
$a^n b^m c^{n+m}$	CFL	*mašina za verifikaciju sabiranja*
$a^n b^m c^{nm}$	CSL	*mašina za verifikaciju množenja*

Zadatak

Kako bi izgledala CFG gramatika i/ili PDA za jezike sa prethodnog slajda +

- a^nbc^n
- $a^{2n}b^n$
- $a^nb^ma^n$

Dokazati da su unija, konkatenacija i zvezda za dva proizvoljna CF isto CF jezici odnosno da su CF jezici zatvoreni za ove operacije.

Zamislite primere; na primer palindrom i a^nb^n.

Šta je sa presekom i komplementom?

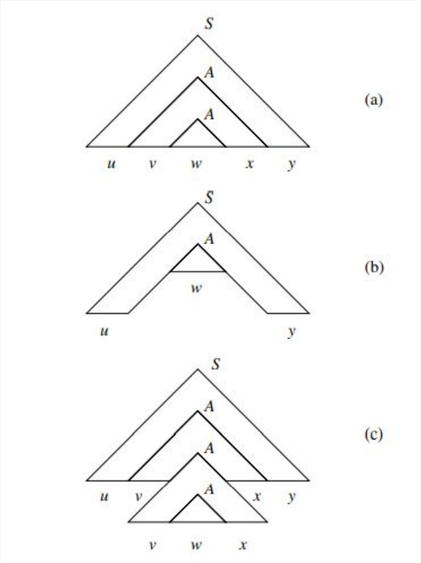
presek - kontraprimer ($a^nb^nc^n$ je presek koja dva CF jezika?)

komplement - pp suprotno; komplement unije komplemenata

Lema o napumpavanju za CFL

Neka je L konteksto slobodan jezik .

Tada postoji broj n takav da za sve reči r ∈ L takve da |r| >= n postoji podstring reči r čija je dužina manja od n koji sadrži pumpu.



r = uvwxy

|vx|>= 1 (bar jedan nije prazan)

|vwx| <= n

$uv^iwx^iz \in L$, i >= 0

n je konstanta napumpavanja jezika L.