

Primer 1: Za gramatiku zadatu sledećim skupom smena:

- (1) $Blok \rightarrow \mathbf{begin} \textit{NizNar} \mathbf{end}$
- (2) $\textit{NizNar} \rightarrow \textit{NizNar} \ ; \ \textit{Naredba}$
- (3) $\textit{NizNar} \rightarrow \textit{Naredba}$
- (4) $\textit{Naredba} \rightarrow \textit{Dodela}$
- (5) $\textit{Naredba} \rightarrow \textit{Blok}$
- (6) $\textit{Dodela} \rightarrow \mathbf{ID} := \textit{Izraz}$
- (7) $\textit{Izraz} \rightarrow \textit{Izraz} + \mathbf{CONST}$
- (8) $\textit{Izraz} \rightarrow \mathbf{CONST}$

kreirati LR sintaksnu tabelu i, korišćenjem kreirane tabele, proveriti da li je sledeći blok ispravno napisan:

```
begin
    ID := CONST
end
```

Rešenje:

1. Kreiranje kanoničkog skupa LR pravila.

Uvodimo smenu $B' \rightarrow B$ i definišemo prvo zatvaranje skupa LR pravila:

$$l_0: \quad B' \rightarrow \cdot B \quad (0,1)$$

Tačka koja razdvaja vidljivi prefiks od neprepoznatog dela smene se nalazi ispred neterminalnog simbola što znači da u nastavku analize treba prepoznati neterminalni simbol. Neterminalni simboli se ne pojavljuju u ulaznom kodu pa prepoznavanje neterminalnog simbola podrazumeva, u stvari, prepoznavanje desne strane neke od smena na čijoj se levoj strani nalazi taj simbol. Zbog toga se ovom zatvaranju dodaje skup pravila izvedenih iz smena za preslikavanje tog neterminalnog simbola sa tačkom na početku. U ovom slučaju dodajemo pravilo:

$$B \rightarrow \cdot \mathbf{begin} \textit{NN} \mathbf{end} \quad (0,2)$$

Kako smo rekli da tačka razdvaja prepoznati deo (vidljivi prefiks) od neprepoznatog, to znači da, kada se analizator nađe u posmatranom stanju, u nastavku analize očekujemo pojavu simbola koji se nalaze iza tačaka u pravilima koja pripadaju tekućem zatvaranju. U konkretnom primeru, simboli koji se očekuju u stanju l_0 su B i **begin**. To znači da treba definisati 2 prelaza iz zatvaranja l_0 – kada se pojavi simbol B i kada se pojavi simbol **begin**.

$$l_1 = \textit{goto}(l_0, B) :$$

Prepoznat je simbol B , vidljivi prefiks označen u pravilu (0,1) se povećava – tačka se pomera iza prepoznatog simbola pa ovom zatvaranju pripada pravilo

$$B' \rightarrow B \cdot$$

Pošto se tačka našla na kraju pravila, ovo stanje je redukciono stanje za smenu iz koje je pravilo izvedeno (tj. za smenu $B' \rightarrow B$). Pošto je ovo pomoćna (fiktivna) smena, ova redukcija se ne vrši, već kad analizator dođe do ovog stanja, ukoliko je ceo ulazni kod analiziran, ulazni kod se prihvata (tj. korektan je).

$l_2 = goto(l_0, \mathbf{begin}) :$

Prepoznat je simbol **begin**, vidljivi prefiks označen u pravilu (0,2) se povećava – tačka se pomera iza prepoznatog simbola pa ovom zatvaranju pripada pravilo

$$B \rightarrow \mathbf{begin} . NN \mathbf{end}$$

Tačka se nalazi ispred neterminalnog simbola NN , pa ovom zatvaranju pripadaju i pravila:

$$NN \rightarrow . NN ; N$$

$$NN \rightarrow . N$$

Pošto ovom zatvaranju pripada i pravilo u kojem je tačka se ispred neterminalnog simbola N , ovom zatvaranju treba dodati i pravila:

$$N \rightarrow . D$$

$$N \rightarrow . B$$

Sada se uočavaju pravila u kojima je tačka ispred neterminalnog simbola D i B pa ovom zatvaranju treba dodati i pravila:

$$D \rightarrow . \mathbf{ID} := I$$

$$B \rightarrow . \mathbf{begin} NN \mathbf{end}$$

$l_3 = goto(l_2, NN) :$

Stanju l_2 pripadaju 2 pravila u kojima je tačka ispred simbola NN pa će ovom zatvaranju pripadati pravila:

$$B \rightarrow \mathbf{begin} NN . \mathbf{end}$$

$$NN \rightarrow NN . ; N$$

$l_4 = goto(l_3, \mathbf{end}) : \quad B \rightarrow \mathbf{begin} NN \mathbf{end} .$

Ovom zatvaranju pripada pravilo sa tačkom na kraju. To pravilo je izvedeno iz prve smene gramatike. To znači da će stanje koje odgovara ovom zatvaranju biti redukciono stanje za smenu 1.

$l_5 = goto(l_3, ;) : \quad NN \rightarrow NN ; . N$

$$N \rightarrow . D$$

$$N \rightarrow . B$$

$$D \rightarrow . \mathbf{ID} := I$$

$$B \rightarrow . \mathbf{begin} NN \mathbf{end}$$

$l_6 = goto(l_5, N) : \quad NN \rightarrow NN ; N . \quad - \text{ redukciono stanje za smenu 2}$

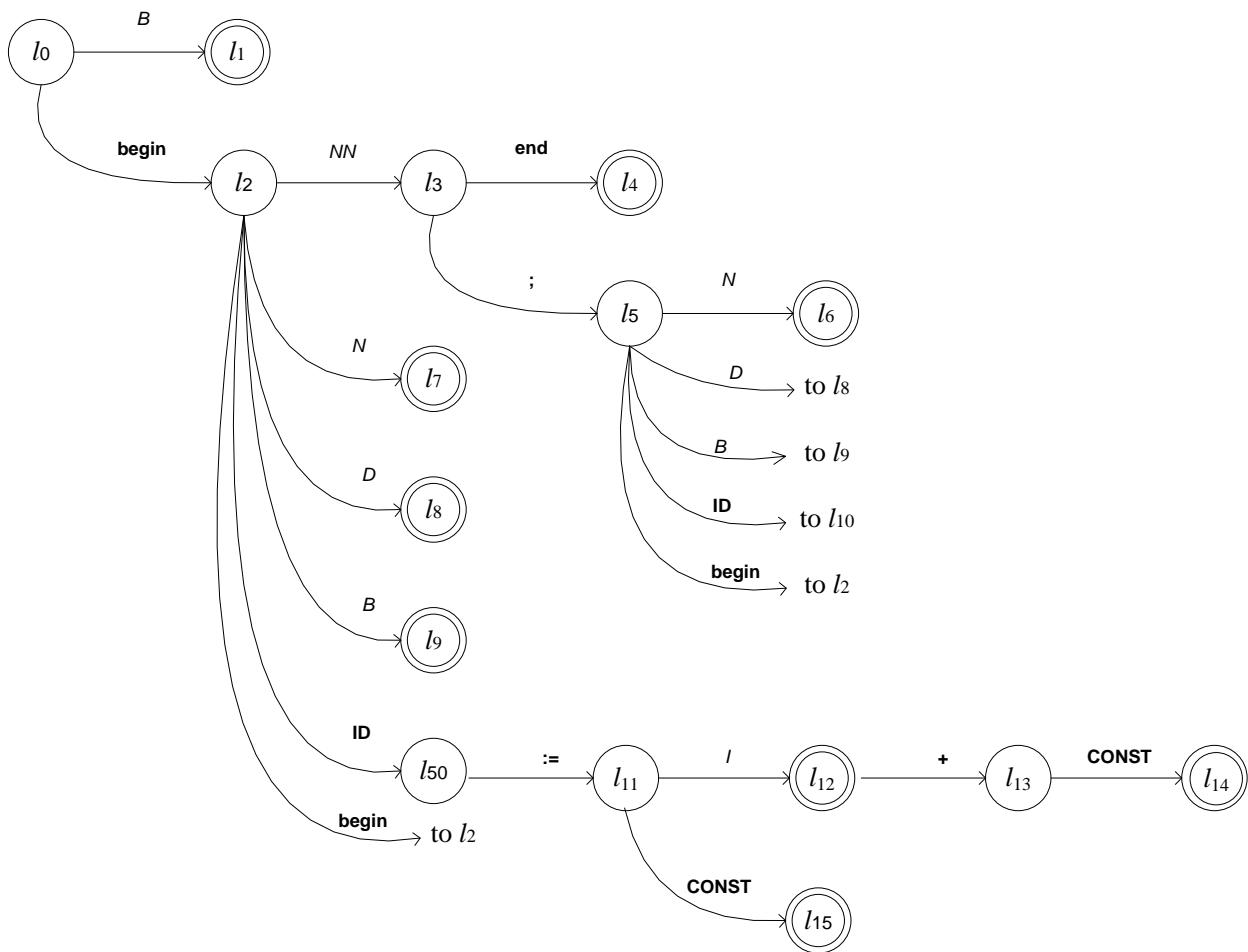
$l_7 = goto(l_2, N) : \quad NN \rightarrow N . \quad - \text{ redukciono stanje za smenu 3}$

$l_8 = \text{goto}(l_2, D) : \quad N \rightarrow D. \quad \text{- redukcionno stanje za smenu 4}$
 $l_9 = \text{goto}(l_2, B) : \quad N \rightarrow B. \quad \text{- redukcionno stanje za smenu 5}$
 $l_{10} = \text{goto}(l_2, \mathbf{ID}) : \quad D \rightarrow \mathbf{ID} . := I$
 $l_{11} = \text{goto}(l_{10}, :=) : \quad D \rightarrow \mathbf{ID} := . I$
 $\quad \quad \quad I \rightarrow . I + \mathbf{CONST}$
 $\quad \quad \quad I \rightarrow . \mathbf{CONST}$
 $l_{12} = \text{goto}(l_{11}, I) : \quad D \rightarrow \mathbf{ID} := I . \quad \text{- redukcionno stanje za smenu 6}$
 $\quad \quad \quad I \rightarrow I . + \mathbf{CONST}$
 $l_{13} = \text{goto}(l_{12}, +) : \quad I \rightarrow I + . \mathbf{CONST}$
 $l_{14} = \text{goto}(l_{13}, \mathbf{CONST}) : \quad I \rightarrow I + \mathbf{CONST} . \quad \text{- redukcionno stanje za smenu 7}$
 $l_{15} = \text{goto}(l_{11}, \mathbf{CONST}) : \quad I \rightarrow \mathbf{CONST} . \quad \text{- redukcionno stanje za smenu 8}$
 $\text{goto}(l_2, \mathbf{begin}) = l_2$
 $\text{goto}(l_5, D) = l_8$
 $\text{goto}(l_5, B) = l_9$
 $\text{goto}(l_5, \mathbf{ID}) = l_{10}$
 $\text{goto}(l_5, \mathbf{begin}) = l_2$

2. Crtanje grafa prelaza konačnog automata za prepoznavanje vidljivih prefiksa

Za svako zatvaranje skupa LR pravila crtamo po jedan čvor u grafu. Potezi u grafu su određeni prelazima koje smo definisali prilikom kreiranja ovog skupa LR pravila. Npr. zatvaranje l_1 smo definisali kao prelaz iz stanja l_0 kad se pojavi simbol B , pa u grafu postoji grana između stanja l_0 i l_1 označena simbolom B .

Redukciona stanja se obeležavaju kao završna stanja u automatu.



3. Popunjavanje LR sintaksne tabele

- Upis *shift* akcija

Svakom potegu koji je označen terminalnim simbolom odgovara po jedna *shift* akcija. Npr. stanja l_0 i l_2 su povezana potegom koji je obeležen terminalnim simbolom **begin**. To znači da će *akcija*(0,**begin**) biti s2. Stanja l_3 i l_4 su povezana potegom koji je obeležen terminalnim simbolom **end** pa je *akcija*(3,**end**)=s4, itd.

- Popunjavanje prelaza u LR tabeli

Svakom potegu koji je označen neterminalnim simbolom odgovara po jedan prelaz u LR sintaksoj tabeli. Npr. stanja l_0 i l_1 su povezana potegom koji je obeležen neterminalnim simbolom P . To znači da je *prelaz*(0, P)=1, itd.

- Upis reduce akcija:

Reduce akcije se pišu u vrstama koje odgovaraju završnim stanjima automata. Npr. završno stanje l_4 sadrži redukciono pravilo:

$$B \rightarrow \text{begin } NN \text{ end} .$$

izvedeno iz smene 1. (Redukciono pravilo je pravilo koje sadrži tačku na kraju.) To znači da se u stanju 4 vrši redukcija po smeni 1 kada se na ulazu pojavi bilo koji simbol koji pripada FOLLOW funkciji za neterminalni simbol koji se nalazi na levoj strani smene, tj. za simbol B . Kako je $\text{FOLLOW}(B) = \{ \# , \text{end} , ; \}$,

$$akcija(4, \#) = akcija(4, \mathbf{end}) = akcija(4, ;) = r1.$$

Slično se definišu i ostale *reduce* akcije. FOLLOW funkcije za ostale neterminalne simbole su:

$$FOLLOW(NN) = \{ \mathbf{end}, ; \}$$

$$FOLLOW(N) = \{ \mathbf{end}, ; \}$$

$$FOLLOW(D) = \{ \mathbf{end}, ; \}$$

$$FOLLOW(I) = \{ +, \mathbf{end}, ; \}$$

Jedino završno stanje u kojem se ne vrši redukcija je stanje 1. Rečeno je već da se redukcija po novouvedenoj pomoćnoj smeni ne vrši, već ako je cela ulazna datoteka pročitana (tj. sledeći simbol je #), kod je kompletno ispravan pa je $akcija(1, \#) = acc$.

	AKCIJE								PRELAZI				
	begin	end	;	ID	:=	+	CONST	#	B	NN	N	D	I
0	s2								1				
1								acc					
2	s2			s10					9	3	7	8	
3		s4	s5										
4		r1	r1					r1					
5	s2			s10					9		6	8	
6		r2	r2										
7		r3	r3										
8		r4	r4										
9		r5	r5										
10					s11								
11							s15						12
12		r6	r6			s13							
13							s14						
14		r7	r7			r7							
15		r8	r8			r8							

4. Analiza ulaznog niza

```

begin
    ID := CONST
end

```

Radni magacin (α)	Next	Akcija koja će se izvršiti
0	begin	s2
0 begin 2	ID	s10
0 begin 2 ID 10	:=	s11
0 begin 2 ID 10 := 11	CONST	s15
0 begin 2 ID 10 := 11 CONST 15	end	r8 (smena 8: $I \rightarrow \mathbf{CONST}$ prelaz(11, I) = 12)
0 begin 2 ID 10 := 11 I 12	end	r6 (smena 6: $D \rightarrow \mathbf{ID} := I$ prelaz(2, D) = 8)
0 begin 2 D 8	end	r4 (smena 4: $N \rightarrow D$ prelaz(2, N) = 7)
0 begin 2 N 7	end	r3 (smena 3: $NN \rightarrow N$ prelaz(2, NN) = 3)
0 start 2 NN 3	end	s4
0 start 2 NN 3 end 4	#	r1 (smena 1: $B \rightarrow \mathbf{begin} \mathbf{NN} \mathbf{end}$ prelaz(0, B) = 1)
0 B 1	#	acc - Ulazni kod je prepoznat

Primer 2: Formirati LR sintaksnu tabelu gramatike koja je zadata skupom smena:

$E \rightarrow E+I \mid I$

$I \rightarrow a(E) \mid a$

Rešenje:

1. Kanonički skup LR pravila:

l_0 :
 $E' \rightarrow .E$
 $E \rightarrow .E+I$
 $E \rightarrow I.$
 $I \rightarrow .a(E)$
 $I \rightarrow .a$

$l_4 = \text{goto}(l_0, I)$: $E \rightarrow I.$

$l_5 = \text{goto}(l_0, a)$: $I \rightarrow a.(E)$
 $I \rightarrow a.$

$l_1 = \text{goto}(l_0, E)$: $E' \rightarrow E.$
 $E \rightarrow E.+I$

$l_6 = \text{goto}(l_5, ($: $I \rightarrow a.(E)$
 $E \rightarrow .E+I$
 $E \rightarrow .I$
 $I \rightarrow .a(E)$
 $I \rightarrow .a$

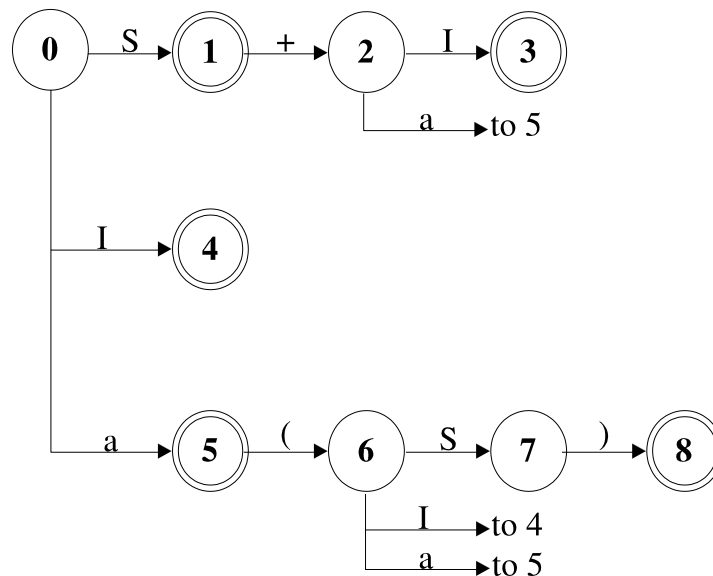
$l_2 = \text{goto}(l_1, +)$: $E \rightarrow E+.I$
 $I \rightarrow .a(E)$
 $I \rightarrow .a$

$l_7 = \text{goto}(l_6, E)$: $I \rightarrow a(E.)$
 $E \rightarrow E.+I$

$l_3 = \text{goto}(l_2, I)$: $E \rightarrow E+I.$

$l_8 = \text{goto}(l_7,)$: $I \rightarrow a(E).$

2. Graf prelaza konačnog automata za prepoznavanje vidljivih prefiksa:



LR sintaksna tabela:

	akcije					prelazi	
	a	+	()	#	S	I
0	s5					1	4
1		s2			acc		
2	s5						3
3		r1		r1	r1		
4		r2		r2	r2		
5		r4	s6	r4	r4		
6	s5					7	4
7		s2		s8			
8		r3		r3	r3		