

Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva

**Seminarski rad iz predmeta  
Prevođenje programskih jezika**

Zadatak broj 2052

Zagreb, prosinac 2009.

# Seminarski rad iz predmeta Uvod u teoriju računarstva

**Student:**

**Matični broj studenta:** 0036437142

**Zadatak broj 2052:** Programski ostvariti **generator potisnog automata** iz zadane Q gramatike.

# Uvod

Parsiranje je postupak prepoznavanja niza i gradnja generativnog stabla na temelju zadanih produkcija kontekсно neovisne gramatike. Parsiranje od vrha prema dnu započinje gradnju generativnog stabla vrhom generativnog stabla koji je označen početnim nezavršnim znakom gramatike. Gradnja stabla se nastavlja primjenom produkcija gramatike na čvorove generativnog stabla koji su označeni nezavršnim znakovima gramatike. Tijekom parsiranja znakovi se čitaju s lijeve nadesno, te se na temelju znaka i znakova lijeve i desne strane produkcije primjeni odgovarajuće strana produkcije. Da bi neka kontekсно neovisna gramatika  $G$  bila  $Q$ -gramatika mora zadovoljiti slijedeće uvijete:

- Desna strana bilo koje produkcije mora započeti završnim znakom gramatike ili je ta desna strana produkcije prazan niz  $\varepsilon$ .
- U slučaju da više produkcije ima isti nezavršni znak na lijevoj strani, onda njihovi skupovi *PRIMIJE* ne smiju imati zajedničke elemente.

Za zadanu  $Q$ -gramatiku  $G = (V, T, P, S)$  moguće je izgraditi deterministički potisni automat  $M = (\{q_0\}, \Sigma, \Gamma, \delta, q_0, Z_0, \emptyset)$  koji prihvaća praznim stogom. Pošto je konstrukcija potisnog automata prilagođena programskom ostvarenju, dodani su posebni znakovi za označavanje kraja ulaznog niza  $\perp$  i znaka dna stoga  $\nabla$ . Za izgrađeni potisni automat vrijede slijedeća pravila:

- Potisni automat ima samo jedno stanje koje je ujedno i početno stanje,  $Q = \{q_0\}$ .
- Skup ulaznih znakova potisnog automata jednak je skupu znakova gramatike uvećan oznakom kraja niza  $\perp$ ,  $\Sigma = T \cup \{\perp\}$ .
- Skup znakova stoga potisnog automata čine oznaka dna stoga  $\nabla$ , skup nezavršnih znakova gramatike  $V$  i skup svih onih završnih znakova gramatike  $T'$  koji su na desnim stranama produkcija, ali ne isključivo na krajnje lijevim mjestima,  $\Gamma = \{\nabla\} \cup V \cup T'$ .
- Na početku rada na stogu se nalazi oznaka dna stoga  $\nabla$  i početni nezavršni znak  $S$ .
- Funkcije prijelaza potisnog automata definira se dvodimenzionalnom tablicom. Redci tablice su znakovi stoga, a stupci ulazni znakovi. Elementi tablice označavaju akciju potisnog automata na ulaznom nizu i akciju na stogu. Tablica se popunjava na slijedeći način:
  - Ako je produkcija oblika:  $A \rightarrow b\alpha$  gdje je  $A$  nezavršni znak,  $b$  završni znak gramatike, a  $\alpha$  preostali niz završnih i nezavršnih znakova, u tablicu se ubacuje akcija potisnog automata:

*Zamijeni( $\alpha'$ );*  
*Pomakni;*

u redak koji je označen nezavršnim znakom  $A$  i završnim znakom  $b$ .

- Ako je produkcija oblika  $A \rightarrow b$  onda se na u tablicu ubacuje akcija:

*Izvuci;*  
*Pomakni;*

- Ako je produkcija oblika  $A \rightarrow \varepsilon$  onda se u redak tablice A i sve stupce određene znakovima skupa PRIMJENI ( $A \rightarrow \varepsilon$ ) zapisuju akcije:

*Izvuci;*  
*Zadrži;*

- Ako je za nezavršni znak A zadana  $\varepsilon$ -produkcija i ako u prethodnim koracima nije popunjen neki od elemenata u retku A, onda se za taj element tablice definira akcija:

*Izvuci;*  
*Zadrži;*

- Ako je završni znak b znak stoga, element tablice koji je u stupcu b i retku b određuje slijedeće akcije potisnog automata:

*Izvuci;*  
*Pomakni;*

- Element tablice koji je označen oznakom dna stoga  $\nabla$  i oznakom kraja ulaznog niza  $\perp$  određuje akciju prihvatanja niza:

*Prihvati;*

- Svi ostali nepopunjeni elementi tablice popunjavaju se oznakom neprihvatljivosti niza:

*Odbaci;*

Akcija *Zamijeni*( $\alpha'$ ) nezavršni znak A koji je u tom trenutku na vrhu stoga zamijeni nizom  $\alpha$ . Znakovi niza  $\alpha$  stavljaju se na stog obrnutim redoslijedom od redoslijeda koji su navedeni u produkciji. Akcija *Pomakni* miče glavu za čitanje potisnog automata na slijedeći znak u ulaznom nizu, dok akcija *Izvuci* uzima sa stoga nezavršni znak koji je u tom trenutku na vrhu stoga.

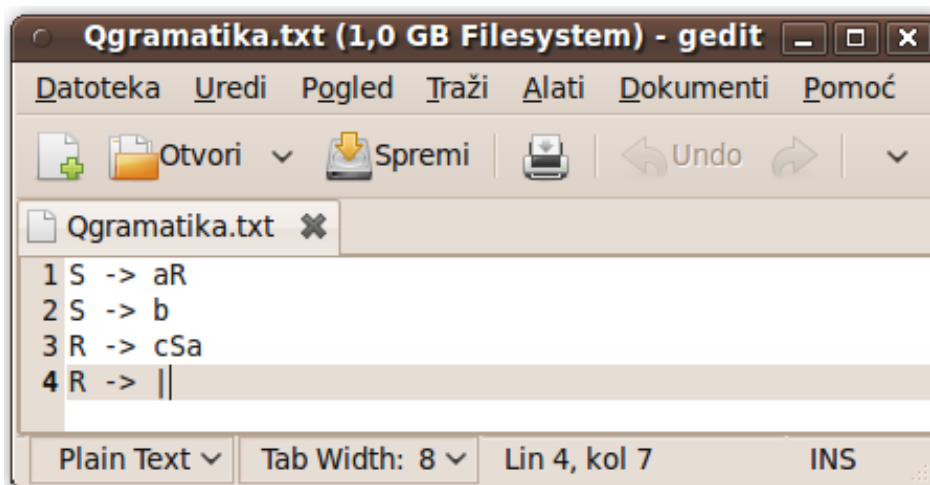
# Ostvarenje

Generator potisnog automata iz zadane Q gramatike ostvaren je u skriptnom programskom jeziku Python 3.1.

Generator se može razložiti na četiri osnovne cjeline:

- cjelina za čitanje iz datoteke i uklanjanje nepotrebnih praznina
- cjelina za pretraživanje završnih i nezavršnih znakova u zadanoj gramatici
- cjelina za popunjavanje tablice funkcija prijelaza potisnog automata
- cjelina za formatirani ispis tablice

Generator očekuje ulaznu datoteku "Qgramatika.txt" u istom direktoriju generatora. Ulazna datoteka može imati ovakav izgled:



Ilustracija 1: Ulazna datoteka "Qgramatika.txt"

gdje su znakovi pisani velikim slovom nezavršni znakovi gramatike a svi ostali znakovi osim '-', '|', '⊥' i '\$' su završni znakovi gramatike. Generator koristi znak '-'>' kako bi mogao prepoznati lijevu i desnu stranu produkcije. Znak '|' u ulaznoj datoteci predstavlja  $\epsilon$ -produkciju, dok znakovi '⊥' i '\$' predstavljaju oznaku kraja ulaznog niza i oznaku kraja stoga. Znakovi '⊥' i '\$' se ne koriste u ulaznoj datoteci.

Generator radi na principu da u jednu listu koja predstavlja tablicu prvotno popuni sve elemente tablice akcijama *Odbaci*; . Zatim za svaku produkciju provjeri tip produkcije, da li je produkcija oblika  $A \rightarrow ba$ ,  $A \rightarrow b$ ,  $A \rightarrow \epsilon$  ili  $A \rightarrow ab$ .

Ako je produkcija tipa  $A \rightarrow ba$  te postoji nezavršni znak A i završni znak b, onda se na tu lokaciju u tablicu upisuje redni broj produkcije *Zamijeni(a')*; kako bi pri ispisu tablice svi stupci bili jednake širine.

Ako je produkcija oblika  $A \rightarrow b$  te postoji nezavršni znak A i završni znak b, onda se u tablicu stavlja *Izvuci; Pomakni*;

Ako je produkcija tipa  $A \rightarrow \epsilon$ , tada se u posebnu listu stavlja nezavršni znak A. Nakon što su obrađene sve produkcije, iz te liste se za sve nezavršne znakove u tablicu zamjenjuje *'Odbaci's 'Izvuci; Zadrzi'*;

Ako je produkcija oblika  $A \rightarrow ab$ , tj u produkciji se nalazi završni znak na krajnjoj desnoj strani desne produkcije, tada se element tablice s stupcem i retkom završnog znaka popunjava s *Izvuci; Pomakni;*.

Izgled ispisa generirane tablice prijelaza potisnog automata prikazano je slijedećom slikom:

```

Python 3.1.1+ (r311:74480, Nov  2 2009, 15:45:00)
[GCC 4.4.1] on linux2
Type "copyright", "credits" or "license()" for more information.
==== No Subprocess ====
>>>
Ulazna gramatika: [('S', 'aR'), ('S', 'b'), ('R', 'cSa'), ('R', '|')]

          a          b          c          |
R      Izvuci; Zadrzi; |Izvuci; Zadrzi; |(1)      |Izvuci; Zadrzi; |
S      (2)      |Izvuci; Pomakni;|Odbaci      |Odbaci      |
a      Izvuci; Pomakni;|Odbaci      |Odbaci      |Odbaci      |
$      Odbaci      |Odbaci      |Odbaci      |Prihvati      |

(1) =>  Zamijeni('aS'); Pomakni;
(2) =>  Zamijeni('R'); Pomakni;

Pritisni 'Enter' za kraj...

```

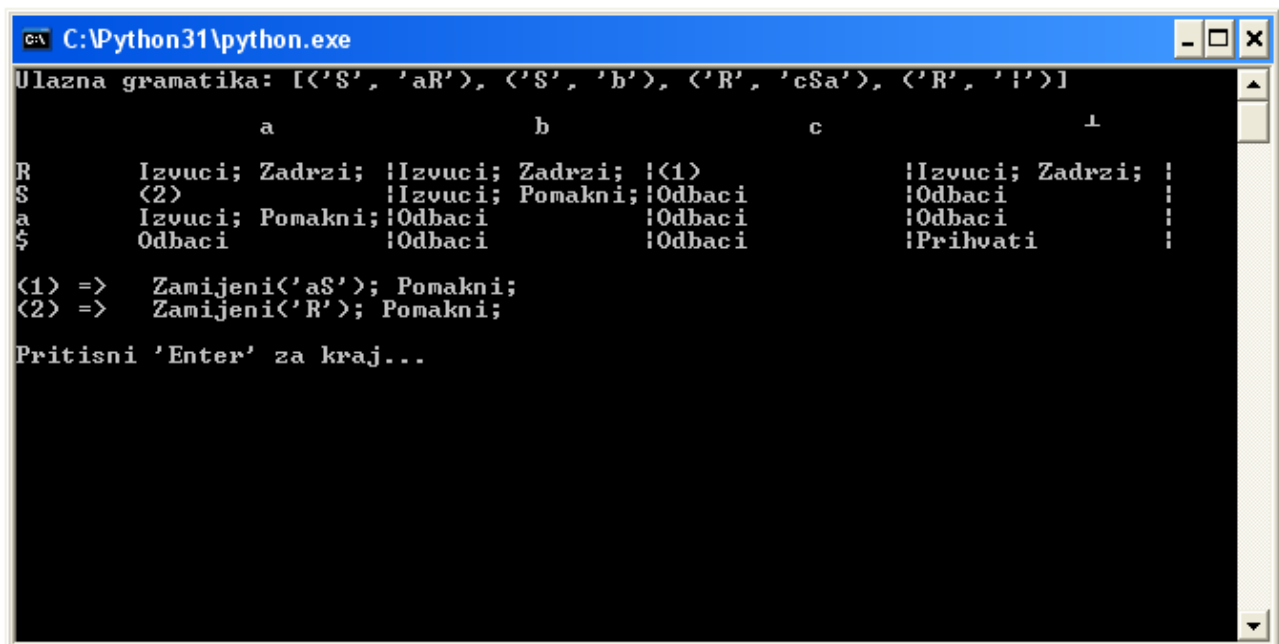
*Ilustracija 2: Izgrađena tablica prijelaza potisnog automata*

## Prevođenje i pokretanje:

Pošto je generator pisan u Pythonu, koji je skriptni programski jezik, prevođenje u izvršnu datoteku nije moguće. Već za pokretanje potrebno je imati Pythonov interpreter, koji se može nabaviti ovdje: [Python 3.1.1](#) (Windows) , [Python 3](#) (Linux, Ubuntu) Također, Python 3.x nije kompatibilan s prijašnjim verzijama Pythona poput verzija 1.x i 2.x, pa stoga stariji interpreteri neće ispravno prikazati rezultate.

Izvorni kod se može pogledati otvaranjem datoteke "SSP\_zadatak2052.py" u Pythonovoj razvojnoj okolini, koja dolazi zajedno u instalacijskom paketu s Python interpreteru, ili otvaranjem u nekom tekst pregledniku.

U Windows okruženju, nakon što je instaliran Python interpreter, dvoklikom na datoteku izvornog koda "SSP\_zadatak2052.py" pokreće se interpreter. Rezultati se prikazuju u konzoli.



```
C:\Python31\python.exe
Ulazna gramatika: [(<'S', 'aR'), (<'S', 'b'), (<'R', 'cSa'), (<'R', 'l')]
```

	a	b	c	l
R	Izvuci; Zadrzi; !Izvuci; Zadrzi; !(<1>			!Izvuci; Zadrzi; !
S	<2>	!Izvuci; Pomakni; !Odbaci		!Odbaci
a	Izvuci; Pomakni; !Odbaci		!Odbaci	!Odbaci
\$	Odbaci	!Odbaci	!Odbaci	!Prihvati

```
<1> => Zamijeni(<'aS')>; Pomakni;
<2> => Zamijeni(<'R')>; Pomakni;
Pritisni 'Enter' za kraj...
```

Ilustracija 3: Prikazani rezultati u Windows konzoli

U Unix okruženju pokretanje se vrši preko naredbe `python3 SSP_zadatak2052.py`. Prije pokretanja potrebno se je pozicionirati u mapu generatora. Također, pokretanje se može izvršiti iz Pythonovog razvojnog sučelja (Python IDLE).

```
marko@Zvjerr: /media/946C-8FA1/SSP_PPJ
Datoteka  Uredi  Pogled  Terminal  Pomoć
marko@Zvjerr:/media/946C-8FA1/SSP_PPJ$ python3 SSP_zadatak2052.py
Ulazna gramatika: [('S', 'aR'), ('S', 'b'), ('R', 'cSa'), ('R', '|')]

          a          b          c          ↓
R      Izvuci; Zadrzi; |Izvuci; Zadrzi; |(1)      |Izvuci; Zadrzi; |
S      (2)      |Izvuci; Pomakni; |Odbaci      |Odbaci      |
a      Izvuci; Pomakni; |Odbaci      |Odbaci      |Odbaci      |
$      Odbaci      |Odbaci      |Odbaci      |Prihvati      |

(1) =>  Zamijeni('aS'); Pomakni;
(2) =>  Zamijeni('R'); Pomakni;

Pritisni 'Enter' za kraj...
```

*Ilustracija 4: Pokretanje generatora iz "bash" okruženja*



## Zaključak

U ovom seminarskom radu izrađen je generator potisnog automata iz zadane Q gramatike, te je u gornjem tekstu opisan način pokretanja generatora. Generator ne provjerava da li je zadana gramatika Q gramatika.

Izvedba ovog generatora je dodatno olakšana izborom programskog jezika s velikim brojem ugrađenih funkcija za lakši rad s listama i poljima znakova. Navedene funkcije omogućile su čitljiviji, kraći kod i jednostavnije otklanjanje pogrešaka. Jedan od primjera je ispisivanje liste, gdje je potrebna jedna naredba, te se korisnika programera ne opterećuje dodatnim poslom neveteanim za implementaciju algoritma.

Najveća mana generatora je bio formatiranje ispisa tablice u konzolu. Naime ukoliko u zadanoj Q-gramatici postoji više od četiri različita završna znaka, onda je tablica preširoka da bi stala u konzolu, pa je potrebno proširiti konzolu. Jedno od mogućih rješenja toga problema je izrada grafičkog sučelja u kojem bi se prikazivali rezultati.