Programiranje I
Beleške sa vežbi

Smer *Informatika*
Matematički fakultet, Beograd

Sana Stojanović

October 28, 2007

1 Formalni jezici i formalne gramatike

[1]

1.1 Jezici: zadaci

Zadatak 1 *Dokazati da ne postoji nijedna reč x azbuke $\Sigma = \{a, b\}$ za koju važi jednakost $xa = bx$.*

Rešenje:
Izvedimo dokaz matematičkom indukcijom po dužini reči x.

1° Za praznu reč ε ne važi jednakost $\varepsilon a = b\varepsilon$ (jer je $a \neq b$), pa tvrđenje važi za reč x dužine 0.

Za $|x| = 1$, važi $x = a$ ili $x = b$. Međutim, kako je $aa \neq ba$ i $ba \neq bb$, sledi da tvrđenje važi za reči x dužine 1.

2° Pretpostavimo da tvrđenje važi za sve reči dužine $n - 2$ i dokažimo da onda važi i za reči dužine n.

Pretpostavimo suprotno – da postoji reč x dužine n takva da važi $xa = bx$. Dakle, reč x počinje slovom b, a završava se slovom a, pa reč x može biti napisana u obliku $x = bya$, gde je y reč dužine $|x| - 2 = n - 2$. Onda važi:

$$byaa = bbya \ ,$$

[1]Zasnovano na materijalu "Teorija algoritama jezika i automata" Predraga Janičića

odakle nakon skraćivanja sledi

$$ya = by \ ,$$

tj. reč y je rešenje zadate jednačine i $|y| = n - 2$, što je u kontradikciji sa induktivnom pretpostavkom, odakle sledi da ne postoji reč x dužine n takva da važi $xa = bx$.

Tvrđenje je dokazano za reči dužine 0 i 1, pa, iz dokazanog induktivnog koraka, sledi da tvrđenje važi za sve prirodne brojeve, čime je dokazano da ne postoji nijedna reč x azbuke $\Sigma = \{a, b\}$ za koju važi jednakost $xa = bx$.

Zadatak 2 *Rešiti nad azbukom* $\Sigma = \{a, b, c\}$ *jednačinu po* x*:* $ax = xa$*.*

Rešenje:
Skup rešenja jednačine je skup reči oblika a^n, $(n \geq 0)$.

($\supseteq$): Za svako $n \geq 0$, reč a^n jeste rešenje, jer $aa^n = a^{n+1} = a^n a$.

($\subseteq$): Indukcijom po dužini reči x dokazujemo da je svako rešenje oblika a^n $(n \geq 0)$.

(1°) Za $|x| = 0$, odnosno $x = \varepsilon$ iz $ax = xa$ sledi $x = a^0$.
Za $|x| = 1$ iz $ax = xa$ sledi $x = a$, tj. $x = a^1$.

(2°) Pretpostavimo da je tvrđenje tačno za sve reči dužine $n-2$ i dokažimo da je tačno i za reči dužine n. Neka je $|x| = n$ i neka je $ax = xa$. Dakle, reč x počinje i završava se slovom a, pa je $x = aya$, gde je y reč nad zadatom azbukom dužine $n-2$. Skraćivanjem se iz $aaya = ayaa$ dobija $ay = ya$, pa je reč y rešenje zadate jednačine dužine $n-2$. Na osnovu induktivne pretpostavke, reč y je oblika a^n, $n \geq 0$, pa je reč x oblika a^{n+2}, $n \geq 0$.

Dakle, svako rešenje date jednačine je oblika a^n $(n \geq 0)$.

Dakle, skup rešenja jednačine je skup reči oblika a^n $(n \geq 0)$.

1.2 Gramatike i jezici: zadaci

Zadatak 3 *Odrediti jezik generisan gramatikom* $G = (N, \Sigma, P, S)$*, gde je* $N = \{S\}$*,*
$\Sigma = \{a, b\}$*,*
$P = \{S \to aS \ \ (1°), S \to b \ \ (2°)\}$*.*

Rešenje:
(Primer izvođenja u gramatici G: $S \overset{1°}{\Rightarrow} aS \overset{1°}{\Rightarrow} aaS \overset{1°}{\Rightarrow} aaaS \overset{2°}{\Rightarrow} aaab$)
Dokažimo da važi:

$$L(G) = \{a^n b \mid n \in \mathbf{N}_0\} \ .$$

$\subseteq$: Svaka završna reč koja se izvodi u gramatici G je oblika $a^n b, n \in \mathbf{N}_0$.

Dokažimo indukcijom jače tvrđenje:

Lema 1 *Sve rečenične forme koje mogu biti izvedene u gramatici G su oblika $a^n S$ ili oblika $a^n b$ ($n \in \mathbf{N}_0$).*

Dokaz indukcijom po dužini izvođenja:

(1) $k = 0$: U izvođenju nije primenjeno nijedno pravilo izvođenja, tj. izvođenje je trivijalno; za početni simbol S, dakle, dobija se takođe reč S. Kako je $S = a^0 S$, tvrđenje važi za $k = 0$.

(2) Pretpostavimo da tvrđenje važi za sva izvođenja čija je dužina manja od k i dokažimo da važi i za izvođenja dužine k.

Neka se reč w može dokazati u k koraka: $S \overset{k}{\Rightarrow} w$. Tada postoji reč w' za koju važi $S \overset{k-1}{\Rightarrow} w' \Rightarrow w$. Reč w' je izvedena izvođenjem dužine $k-1$, pa je, na osnovu induktivne hipoteze, w' oblika $a^n S$ ili oblika $a^n b$ ($n \in \mathbf{N}_0$). Reč w se netrivijalno (izvođenjem dužine 1) izvodi iz reči w', pa w' mora da ima nezavršnih simbola. Dakle, w' je oblika $a^n S$. Ako je u $k-$tom koraku primenjeno pravilo $1°$, onda je w oblika $a^{n+1} S$, a ako je primenjeno pravilo $2°$, onda je w oblika $a^n b$.

Dakle, sve reči koje mogu biti izvedene u gramatici G su oblika $a^n S$ ili oblika $a^n b$ ($n \in \mathbf{N}_0$), što je i trebalo dokazati.

Na osnovu leme, svi članovi izvođenja (sve reči koje mogu biti izvedene u gramatici G) su oblika $a^n S$ ili oblika $a^n b$ ($n \in \mathbf{N}_0$), pa i završne reči – reči bez nezavršnih simbola. Završne reči ne mogu biti oblika $a^n S$ (jer je S nezavršni simbol), pa su sve završne reči oblika $a^n b$ ($n \in \mathbf{N}_0$). Dakle, $L(G) \subseteq \{a^n b \mid n \in \mathbf{N}_0\}$.

$\supseteq$: Svaka reč $a^n b$, $n \in \mathbf{N}_0$, može biti izvedena u gramatici G.

Za proizvoljno n ($n \in \mathbf{N}_0$) reč $a^n b$ može biti izvedena u gramatici G:

$$S \underbrace{\overset{1°}{\Rightarrow} aS \overset{1°}{\Rightarrow} \ldots \overset{1°}{\Rightarrow}}_{n} a^n S \overset{2°}{\Rightarrow} a^n b$$

Zadatak 4 *Odrediti jezik generisan gramatikom $G = (N, \Sigma, P, S)$, gde je $N = \{S\}$,*
$\Sigma = \{a, b\}$,
$P = \{S \to aSb\ \ (1°), S \to \varepsilon\ \ (2°)\}$.

Rešenje:
$W = \{a^n b^n | n \geq 0\}$

Zadatak 5 *Odrediti gramatiku koja generiše jezik $W = \{a^{2i} b^i \mid i > 0\}$.*

Rešenje:

$G = (N, \Sigma, P, S)$, gde je $N = \{S\}$,
$\Sigma = \{a, b\}$,
$P = \{S \to aab \ \ (1^\circ), S \to aaSb \ \ (2^\circ)\}$

Dokažimo da važi $W = L(G)$.

$\subseteq$: Neka je $w \in W$, tj. neka je $w = a^{2i}b^i$, gde je $i > 0$.

<u>w se može izvesti u gramatici G.</u>

$S \overset{2^\circ, i-1}{\Rightarrow} (aa)^{i-1}Sb^{i-1} \overset{1^\circ}{\Rightarrow} (aa)^i b^i$.

Dakle, $W \subseteq L(G)$.

$\supseteq$: Indukcijom se može dokazati da je svaki član izvođenja w oblika $a^{2i}Sb^i$ $(i \geq 0)$ ili oblika $a^{2i}b^i$ $(i > 0)$. Završna reč w ne može da sadrži simbol S, pa je oblika $a^{2i}b^i$ $(i > 0)$, odakle sledi $W \supseteq L(G)$.

Zadaci za vežbu:

Zadatak 6 *Odrediti gramatiku koja generiše jezik* $W = \{a^n b^{[\frac{n}{2}]} \mid n \geq 0\}$.

Rešenje:

$G = (N, \Sigma, P, S)$, gde je $N = \{S\}$,
$\Sigma = \{a, b\}$,
$P = \{S \to aaSb \ \ (1^\circ), S \to \varepsilon \ \ (2^\circ), S \to a \ \ (3^\circ)\}$

Zadatak 7 *Dokazati da ne postoji nijedna reč* y *azbuke* $\Sigma = \{c, d\}$ *za koju važi jednakost* $cy = yd$.

Zadatak 8 *Rešiti nad azbukom* $\Sigma = \{a, b, c\}$ *jednačinu po* y*:* $yb = by$.

Zadatak 9 *Odrediti jezik generisan gramatikom* $G = (N, \Sigma, P, S)$*, gde je* $N = \{S\}$,
$\Sigma = \{a, b\}$,
$P = \{S \to Sb \ \ (1^\circ), S \to a \ \ (2^\circ)\}$.

Zadatak 10 *Odrediti gramatiku koja generiše jezik* $W = \{b^i a^{3i} \mid i \geq 0\}$.

Zadatak 11 *Odrediti gramatiku koja generiše sve palindrome nad azbukom* $\Sigma = \{a, b\}$.

Rešenje:

$G = (N, \Sigma, P, S)$, gde je $N = \{S\}$, $\Sigma = \{a, b\}$,
$P = \{S \to aSa \ \ (1^\circ), S \to bSb \ \ (2^\circ), S \to a \ \ (3^\circ), S \to b \ \ (4^\circ), S \to \varepsilon \ \ (5^\circ)\}$

2 Teorija algoritama

Postoji više formalizama kojima se uvodi pojam izračunljivosti; neki od njih su UR mašine, rekurzivne funkcije, Tjuringove mašine, Postove mašine, Markovljevi algoritmi. Može se dokazati da su klase funkcija izračunljivih funkcija identične za ove formalizme. Cherčova teza tvrdi da je klasa intuitivno, neformalno izračunljivih funkcija identična sa tim, strogo zasnovanim klasama izračunljivih funkcija. U daljem tekstu, pojam izračunljivosti biće uveden i izučavan na bazi UR mašina i rekurzivnih funkcija.

3 UR mašine

UR mašine[2] (URM) su jedan od formalizama za definisanje pojma algoritma. To su apstraktne mašine koje predstavljaju matematičku idealizaciju računara. UR mašina ima neograničen broj registara koje označavamo sa $R_1, R_2, R_3, \ldots$. Svaki od njih u svakom trenutku sadrži neki prirodan broj. Sadržaj k-tog registra (registra R_k) označavamo sa r_k.

R_1	R_2	R_3	$\ldots$
r_1	r_2	r_3	$\ldots$

Sadržaj registara se menja naredbama (instrukcijama) čiji je opis dat u tabeli 3.[3]

oznaka	naredba	dejstvo
$Z(m)$	nula–naredba	$0 \rightarrow R_m$ (tj. $r_m := 0$)
$S(m)$	naredba sledbenik	$r_m + 1 \rightarrow R_m$ (tj. $r_m := r_m + 1$)
$T(m,n)$	naredba prenosa	$r_m \rightarrow R_n$ (tj. $r_n := r_m$)
$J(m,n,p)$	naredba skoka	ako je $r_m = r_n$, idi na p-tu; inače idi na sledeću naredbu

Table 1: Tabela URM naredbi

Izračunavanje na UR mašini karakterišu sledeće osobine:

- URM program P je niz konačno mnogo naredbi ($P : I_1, I_2, \ldots, I_s$).
- Početnu konfiguraciju čini niz prirodnih brojeva $a_1, a_2, \ldots$ u registrima $R_1, R_2, \ldots$.
- Ako je funkcija koju treba izračunati $f(x_1, x_2, \ldots, x_n)$, onda se podrazumeva da su vrednosti $x_1, x_2, \ldots, x_n$ redom smeštene u prvih n registara iza kojih sledi niz nula, kao i da rezultat treba smestiti u prvi registar.

[2]Od engleskog *unlimited register mačine.*

[3]Oznake URM naredbi su uskladjene za nazivima ovih naredbi na engleskom jeziku. Naime, u literaturi na ovom jeziku se koriste termini *zero instruction, succesor instruction, transfer instruction* i *jump instruction.*

- Naredbe se izršavaju sekvencijalno (počevši od prve instrukcije), osim u slučaju naredbe skoka.

- Izračunavanje prestaje samo onda kada ne postoji sledeća naredba koju treba izvršiti.

Definicija 1 *Ako* UR *mašina nakon primene programa* P *na početnu konfiguraciju* $a_1, a_2, \ldots, a_n$ *staje sa radom, onda pišemo*

$$P(a_1, a_2, \ldots, a_n) \downarrow,$$

inače pišemo

$$P(a_1, a_2, \ldots, a_n) \uparrow .$$

Definicija 2 *Ako* UR *mašina nakon primene programa* P *na početnu konfiguraciju* $a_1, a_2, \ldots, a_n$ *staje sa radom i u njenom prvom registru je, kao rezultat izračunavanja, vrednost* b*, onda pišemo*

$$P(a_1, a_2, \ldots, a_n) \downarrow b.$$

Definicija 3 *Neka je* f *parcijalna funkcija*[4] *i* P URM *program. Kažemo da* P izračunava f *ako i samo ako važi*

$$\begin{array}{rl} & (\forall a_1, a_2, \ldots, a_n, b \in \mathbf{N})(P(a_1, a_2, \ldots, a_n) \downarrow b \\ \Leftrightarrow & (a_1, a_2, \ldots, a_n) \in \mathbf{Dom}(f) \wedge f(a_1, a_2, \ldots, a_n) = b). \end{array}$$

Definicija 4 *Funkcija je* URM izračunljiva *ako postoji* URM *program koji je izračunava.*

Osnovna hipoteza teorije algoritama je tzv. teza Cherča.[5] Naglasimo da je ovo tvrdjenje hipoteza, a ne teorema. Naime, ono govori o intuitivnom pojmu algoritma, čija svojstva ne mogu biti formalno ispitivana.

Teza 1 *Klasa intuitivno izračunljivih funkcija identična je sa klasom* URM *izračunljivih funkcija.*

Pojam formalno izračunljivih funkcija uvodi se za:

- UR mašine,

- Tjuringove[6] mašine,

- Postove[7] mašine,

[4]Za funkciju $f : \mathbf{N}^n \rightarrow \mathbf{N}$ kažemo da je *parcijalna* ako je njen domen podskup skupa $\mathbf{N}^n$ ($\mathbf{Dom}(f) \subseteq \mathbf{N}^n$), odnosno kažemo da je *totalna* ako je njen domen $\mathbf{N}^n$ ($\mathbf{Dom}(f) = \mathbf{N}^n$).

[5]Alonzo Church (1903-1995), američki logičar

[6]Alan Turing (1912-1954), britanski matematičar

[7]Emil L. Post (1897-1954), američki matematičar

- rekurzivne funkcije,
- Markovljeve[8] algoritme,
- formalne gramatike.

Može se dokazati da su klase izračunljivih funkcija koje odgovaraju navedenim formalizmima identične. Odatle proističe i sledeća formulacija teze Cherča.

Teza 2 *Klasa intuitivno izračunljivih funkcija identična je sa klasom formalno izračunljivih funkcija.*

Iz teze Cherča (koja je opšte prihvaćena, iako je nedokaziva) sledi da su pojmovi intuitivno izračunljivih funkcija, URM izračunljivih funkcija, Tjuring izračunljivih funkcija itd. ekvivalentni. U daljem tekstu će, jednostavnosti radi, najčešće biti korišćen samo termin *izračunljive funkcije*.

Zadatak 1 *Napisati* URM *program koji izračunava funkciju* $f(x, y) = x + y$.

Rešenje:

Predloženi algoritam se zasniva na sledećoj osobini:

$$x + y = x + \underbrace{1 + 1 + \ldots + 1}_{y}$$

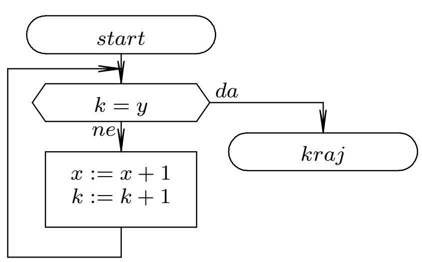
Dakle, vrednosti x se sukcesivno dodaje vrednost 1 y puta. Odgovarajući URM program podrazumeva sledeću početnu konfiguraciju:

R_1	R_2	R_3	$\ldots$
x	y	0	$\ldots$

i sledeću radnu konfiguraciju:

R_1	R_2	R_3	$\ldots$
$x+k$	y	k	$\ldots$

gde $k \in \{0, 1, \ldots, y\}$.



1. $J(3, 2, 100)$
2. $S(1)$
3. $S(3)$
4. $J(1, 1, 1)$

[8] Andrej Andrejevič Markov (1856-1922), ruski matematičar