

Prevodioci i interpretatori - novembar 2005. - teorijski deo

1. Ispitati da li je jezik opisan gramatikom $S \rightarrow xSy|\varepsilon$ regularan. Odgovor dokazati.
2. Konstruisati minimalni deterministički konačni automat koji prihvata regularni jezik $(ab)^*(ab?a|b)^+$
3. LL(1) gramatikom opisati strukturu centralnog dela YACC datoteke u kome je dat opis gramatike bez akcija. Odrediti skupove izbora za datu gramatiku. Tablicom predstaviti odgovarajući potisni automat.
4. Napisati program u C-u koji tehnikom rekurzivnog spusta izračunava vrednost iskazne formule koja se unosi sa standardnog ulaza. Formula se gradi korišćenjem veznika \wedge koji predstavlja konjunkciju, \vee disjunkciju, \Rightarrow implikaciju, i konstanti T i F (veznici su navedeni u opadajućem redosledu prioriteta). Npr. $(T \Rightarrow F) \vee (F \wedge T)$.

Prevodioci i interpretatori - novembar 2005. - teorijski deo

1. Ispitati da li je jezik opisan gramatikom $S \rightarrow xSy|\varepsilon$ regularan. Odgovor dokazati.
2. Konstruisati minimalni deterministički konačni automat koji prihvata regularni jezik $(ab)^*(ab?a|b)^+$
3. LL(1) gramatikom opisati strukturu centralnog dela YACC datoteke u kome je dat opis gramatike bez akcija. Odrediti skupove izbora za datu gramatiku. Tablicom predstaviti odgovarajući potisni automat.
4. Napisati program u C-u koji tehnikom rekurzivnog spusta izračunava vrednost iskazne formule koja se unosi sa standardnog ulaza. Formula se gradi korišćenjem veznika \wedge koji predstavlja konjunkciju, \vee disjunkciju, \Rightarrow implikaciju, i konstanti T i F (veznici su navedeni u opadajućem redosledu prioriteta). Npr. $(T \Rightarrow F) \vee (F \wedge T)$.

Prevodioci i interpretatori - novembar 2005. - teorijski deo

1. Ispitati da li je jezik opisan gramatikom $S \rightarrow xSy|\varepsilon$ regularan. Odgovor dokazati.
2. Konstruisati minimalni deterministički konačni automat koji prihvata regularni jezik $(ab)^*(ab?a|b)^+$
3. LL(1) gramatikom opisati strukturu centralnog dela YACC datoteke u kome je dat opis gramatike bez akcija. Odrediti skupove izbora za datu gramatiku. Tablicom predstaviti odgovarajući potisni automat.
4. Napisati program u C-u koji tehnikom rekurzivnog spusta izračunava vrednost iskazne formule koja se unosi sa standardnog ulaza. Formula se gradi korišćenjem veznika \wedge koji predstavlja konjunkciju, \vee disjunkciju, \Rightarrow implikaciju, i konstanti T i F (veznici su navedeni u opadajućem redosledu prioriteta). Npr. $(T \Rightarrow F) \vee (F \wedge T)$.

Prevodioci i interpretatori - novembar 2005. - teorijski deo

1. Ispitati da li je jezik opisan gramatikom $S \rightarrow xSy|\varepsilon$ regularan. Odgovor dokazati.
2. Konstruisati minimalni deterministički konačni automat koji prihvata regularni jezik $(ab)^*(ab?a|b)^+$
3. LL(1) gramatikom opisati strukturu centralnog dela YACC datoteke u kome je dat opis gramatike bez akcija. Odrediti skupove izbora za datu gramatiku. Tablicom predstaviti odgovarajući potisni automat.
4. Napisati program u C-u koji tehnikom rekurzivnog spusta izračunava vrednost iskazne formule koja se unosi sa standardnog ulaza. Formula se gradi korišćenjem veznika \wedge koji predstavlja konjunkciju, \vee disjunkciju, \Rightarrow implikaciju, i konstanti T i F (veznici su navedeni u opadajućem redosledu prioriteta). Npr. $(T \Rightarrow F) \vee (F \wedge T)$.

Prevodioci i interpretatori - novembar 2005. - teorijski deo

1. Ispitati da li je jezik opisan gramatikom $S \rightarrow xSy|\varepsilon$ regularan. Odgovor dokazati.
2. Konstruisati minimalni deterministički konačni automat koji prihvata regularni jezik $(ab)^*(ab?a|b)^+$
3. LL(1) gramatikom opisati strukturu centralnog dela YACC datoteke u kome je dat opis gramatike bez akcija. Odrediti skupove izbora za datu gramatiku. Tablicom predstaviti odgovarajući potisni automat.
4. Napisati program u C-u koji tehnikom rekurzivnog spusta izračunava vrednost iskazne formule koja se unosi sa standardnog ulaza. Formula se gradi korišćenjem veznika \wedge koji predstavlja konjunkciju, \vee disjunkciju, \Rightarrow implikaciju, i konstanti T i F (veznici su navedeni u opadajućem redosledu prioriteta). Npr. $(T \Rightarrow F) \vee (F \wedge T)$.

Prevodioci i interpretatori - novembar 2005. - praktični deo

1. Direktorijum **studenti** sadrži studentske dosijee imenovane uobičajeno (npr. **mr01678**) koji sadrže datoteke sa zadacima koje su studenti tokom leta radili. Studenti su mogli da biraju jedan od četiri programska jezika i radili su 10 zadataka. Datoteke koje su predali su oblika **broj.ext**, **broj** predstavlja broj zadatka koji je rešavan, a **ext** predstavlja oznaku programskog jezika u kome je zadatak rađen (c, **cpp**, **java**, **pas**). Napisati PERL skript koji na osnovu sadržaja direktorijuma **studenti** kreira HTML tabelu koja pregledno prikazuje rad studenata. Svaka vrsta se odnosi na po jednog studenta i počinje njegovim brojem dosijea, a zatim, za svaki od 10 zadataka, sadrži oznaku jezika u kome je zadatak rađen, odnosno, oznaku – ukoliko student nije radio taj zadatak.

	1	2	3	4	5	6	7	8	9	10	
mr01567	c	c	cpp	java	pas	-	-	cpp	java	pas	[15]
mr01678	c	cpp	java	java	-	c	-	cpp	pas	cpp	

2. Napisati interpretator za mali jezik u kome je definisan tip **string** koji označava niske karaktera. Nad niskama je definisan operator **+** i **+=** koji označavaju konkatenaciju, zatim operator **[m,n]** koji označava izdvajanje *n* karaktera niske počevši od pozicije *m*. Definisan je operator ***** množenja niske brojem *n* koja nadovezuje *n* kopija niske. Definisane su i funkcije **reverse** koja obrće prosledjenu nisku kao i funkcija **print** koja ispisuje nisku na standardni izlaz. Jezik dopušta definisanje promenljivih, a konstante se navode unutar navodnika. Jezik sadrži i operator **=~** koji proverava da li se druga niska pojavljuje kao podniska prve i čiji je rezultat primene niska **found** odnosno **not found**. Npr.

```
string s = "abc";
string s1;
string t = s + "de"*2;
s += "gh";
s1 = s[2,2];
reverse(t);
print(s1);           cg
print(t);           ededcba
```

- (a) Implementirati deklarisanje promenljivih i njihovo inicijalizovanje konstantnim niskama kao i funkciju **print**.

```
string s = "abc";
print(s);
```

- (b) Implementirati operatore konkatenacije **+** i **+=**.

```
string s = "abc" + "def";
s += "ghi";
print(s);
```

- (c) Implementirati množenje niske brojem.

```
string s = "a"*5;
print(s);
```

- (d) Implementirati operator izdvajanja podniski.

```
string s = "abcdefg";
print(s[3,5]);
```

- (e) Implementirati funkciju **reverse**.

```
string s = "abcdefg";
reverse(s);
print(s);
```

- (f) Implementirati funkciju **readln** koja učitava liniju sa standardnog ulaza.

```
string s;
readln(s);
print(s);
```

- (g) Implementirati operator **=~**.

```
string s, t;
readln(s);
readln(t);
print(s =~ t);
```