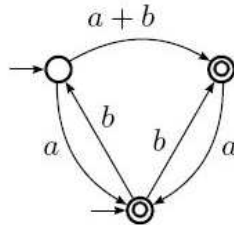


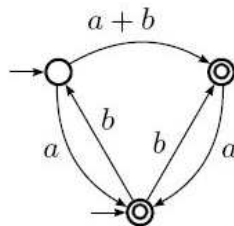
Prevodioci i interpretatori - Januar 2005. - teorijski deo

1. (a) Odrediti minimalni deterministički konačni automat ekvivalentan automatu sa slike.
(b) Regularnim izrazom opisati jezik datog automata.
(c) Koje reči dužine 4 automat ne prepoznaje?



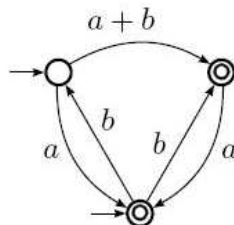
Prevodioci i interpretatori - Januar 2005. - teorijski deo

1. (a) Odrediti minimalni deterministički konačni automat ekvivalentan automatu sa slike.
(b) Regularnim izrazom opisati jezik datog automata.
(c) Koje reči dužine 4 automat ne prepoznaje?



Prevodioci i interpretatori - Januar 2005. - teorijski deo

1. (a) Odrediti minimalni deterministički konačni automat ekvivalentan automatu sa slike.
(b) Regularnim izrazom opisati jezik datog automata.
(c) Koje reči dužine 4 automat ne prepoznaje?



1. Kontekst slobodnom gramatikom opisati deklaraciju funkcije u jeziku C. Na osnovu date kontekst slobodne gramatike konstruisati potisni automat koji metodom sintaksne analize *naviše* prihvata pomenuti jezik. Za nisku

```
int funkcija (int* pa, char str[]);
```

napisati najdešnje izvodjenje u datoj gramatici i nacrtati drvo izvodjenja. Simulirati prihvatanje niske konstruisanim potisnim automatom.

2. Opisati KS-gramatikom jezik uredjenih dvojki. Elementi u uredjenoj dvojci mogu biti iz skupa $\{a, b\}$ ili uredjene dvojke. Na primer, ispravne reči su $(a, (a, (b, a)))$ i $((a, a), (a, (b, b))), (a, b))$. Transformisati tu gramatiku u pravilnu LL(1)-gramatiku (dokazati da jeste LL(1)).
3. Konstruisati konačni transduktor koji brojeve u C-ovskom heksadekadnom zapisu prevodi u ekvivalentan binarni zapis bez vodećih nula (npr. 0h37 se prevodi u 110111, a 0H1AC u 110101100). Potom napisati program koji realizuje dati transduktor (bilo u C-u bilo pomoću lex-a).

1. Kontekst slobodnom gramatikom opisati deklaraciju funkcije u jeziku C. Na osnovu date kontekst slobodne gramatike konstruisati potisni automat koji metodom sintaksne analize *naviše* prihvata pomenuti jezik. Za nisku

```
int funkcija (int* pa, char str[]);
```

napisati najdešnje izvodjenje u datoj gramatici i nacrtati drvo izvodjenja. Simulirati prihvatanje niske konstruisanim potisnim automatom.

2. Opisati KS-gramatikom jezik uredjenih dvojki. Elementi u uredjenoj dvojci mogu biti iz skupa $\{a, b\}$ ili uredjene dvojke. Na primer, ispravne reči su $(a, (a, (b, a)))$ i $((a, a), (a, (b, b))), (a, b))$. Transformisati tu gramatiku u pravilnu LL(1)-gramatiku (dokazati da jeste LL(1)).
3. Konstruisati konačni transduktor koji brojeve u C-ovskom heksadekadnom zapisu prevodi u ekvivalentan binarni zapis bez vodećih nula (npr. 0h37 se prevodi u 110111, a 0H1AC u 110101100). Potom napisati program koji realizuje dati transduktor (bilo u C-u bilo pomoću lex-a).

1. Kontekst slobodnom gramatikom opisati deklaraciju funkcije u jeziku C. Na osnovu date kontekst slobodne gramatike konstruisati potisni automat koji metodom sintaksne analize *naviše* prihvata pomenuti jezik. Za nisku

```
int funkcija (int* pa, char str[]);
```

napisati najdešnje izvodjenje u datoj gramatici i nacrtati drvo izvodjenja. Simulirati prihvatanje niske konstruisanim potisnim automatom.

2. Opisati KS-gramatikom jezik uredjenih dvojki. Elementi u uredjenoj dvojci mogu biti iz skupa $\{a, b\}$ ili uredjene dvojke. Na primer, ispravne reči su $(a, (a, (b, a)))$ i $((a, a), (a, (b, b))), (a, b))$. Transformisati tu gramatiku u pravilnu LL(1)-gramatiku (dokazati da jeste LL(1)).
3. Konstruisati konačni transduktor koji brojeve u C-ovskom heksadekadnom zapisu prevodi u ekvivalentan binarni zapis bez vodećih nula (npr. 0h37 se prevodi u 110111, a 0H1AC u 110101100). Potom napisati program koji realizuje dati transduktor (bilo u C-u bilo pomoću lex-a).