

### Prevodioci i interpretatori - Decembar 2003.

1. (a) Konstruisati konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrže bar tri pojavljivanja simbola 0.  
(b) Konstruisati konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrži bar tri uzastopna pojavljivanja simbola 0  
(c) Koristeći neki od algoritama za konstrukciju *razlike* automata i algoritme *minimizacije* i *determinizacije* ukoliko je potrebno, konstruisati minimalni deterministički konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrže bar 3 pojavljivanja simbola 0, ali ne sadrže 3 uzastopna pojavljivanja simbola 0.
2. (a) Kontekst slobodnom gramatikom opisati deklaraciju strukture u programskom jeziku C. Struktura može da sadži samo polja prethodno deklariranih struktura i osnovnih tipova jezika C. Npr.  
(b) Na osnovu date gramatike konstruisati odgovarajuću LL(1) gramatiku. Odrediti skup izbora za svako pravilo ove gramatike.  
(c) Napisati program u C-u koji simulirajući rad potisnog automata ispituje ispravnost zadate deklaracije strukture. Leksički analizator napisati korišćenjem lex-a.

### Prevodioci i interpretatori - Decembar 2003.

1. (a) Konstruisati konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrže bar tri pojavljivanja simbola 0.  
(b) Konstruisati konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrži bar tri uzastopna pojavljivanja simbola 0  
(c) Koristeći neki od algoritama za konstrukciju *razlike* automata i algoritme *minimizacije* i *determinizacije* ukoliko je potrebno, konstruisati minimalni deterministički konačni automat koji prepoznaje sve one reči nad azbukom  $\Sigma = \{0, 1\}$  koje sadrže bar 3 pojavljivanja simbola 0, ali ne sadrže 3 uzastopna pojavljivanja simbola 0.
2. (a) Kontekst slobodnom gramatikom opisati deklaraciju strukture u programskom jeziku C. Struktura može da sadži samo polja prethodno deklariranih struktura i osnovnih tipova jezika C.  
(b) Na osnovu date gramatike konstruisati odgovarajuću LL(1) gramatiku. Odrediti skup izbora za svako pravilo ove gramatike.  
(c) Napisati program u C-u koji simulirajući rad potisnog automata ispituje ispravnost zadate deklaracije strukture. Leksički analizator napisati korišćenjem lex-a.

3 Data je datoteka koja sadrži tekst nekog C programa. Na osnovu ove datoteke konstruisati HTML datoteku na sledeći način:

- Svakoј liniji C datoteke treba da odgovara jedna linija tela HTML datoteke koja počinje tekućim brojem linije C datoteke i zatim odgovarajućom linijom programa obeleženom odgovarajućim etiketama.
- Sav tekst se piše tako da formatiranje teksta odgovara formatiranju teksta u C datoteci (najbolje je koristiti preformatirani ispis jezika HTML)
- Svaku ključnu reč jezika C obeležiti etiketama

```
<span class="keyword"> i </span>
```

- Sve komentare obeležiti sa

```
<span class="comment"> i </span>
```

- Sve pretprocesorske direktive obeležiti sa

```
<span class="preprocessor"> i </span>
```

- Sve identifikatore ispisati *kurzivom*.
- Sve nedeklarisane promenjive i pozive nedeklarisanih funkcija ispisati **podebljano**.

3 Data je datoteka koja sadrži tekst nekog C programa. Na osnovu ove datoteke konstruisati HTML datoteku na sledeći način:

- Svakoј liniji C datoteke treba da odgovara jedna linija tela HTML datoteke koja počinje tekućim brojem linije C datoteke i zatim odgovarajućom linijom programa obeleženom odgovarajućim etiketama.
- Sav tekst se piše tako da formatiranje teksta odgovara formatiranju teksta u C datoteci (najbolje je koristiti preformatirani ispis jezika HTML)
- Svaku ključnu reč jezika C obeležiti etiketama

```
<span class="keyword"> i </span>
```

- Sve komentare obeležiti sa

```
<span class="comment"> i </span>
```

- Sve pretprocesorske direktive obeležiti sa

```
<span class="preprocessor"> i </span>
```

- Sve identifikatore ispisati *kurzivom*.
- Sve nedeklarisane promenjive i pozive nedeklarisanih funkcija ispisati **podebljano**.