

Prevodioci i interpretatori - Novembar 2003.

1. Proveriti da li su jezici $L_1 = (a^2 + b^*a^+)^*$ i $L_2 = ((a^2)^* + b^*a)^*$ jednaki. Odgovor obavezno dokazati.
2. U programskom jeziku C, tehnikom rekurzivnog spusta implementirati program koji sračunava vrednosti niza logičkih formula zadatih u datoteci čije se ime prosledjuje sa standardnog ulaza. Formule su definisane nad azbukom

$$\Sigma = \{\&, |, =, >, < =, >, T, F, (,)\}$$

pri čemu se svi veznici zapisuju infiksno, sa uobičajenim prioritetom i asocijativnošću. Prilikom sračunavanja istinitosne vrednosti koristiti tehniku lenjog izračunvanja (*lazy evaluation*).

Prevodioci i interpretatori - Novembar 2003.

1. Proveriti da li su jezici $L_1 = (a^2 + b^*a^+)^*$ i $L_2 = ((a^2)^* + b^*a)^*$ jednaki. Odgovor obavezno dokazati.
2. U programskom jeziku C, tehnikom rekurzivnog spusta implementirati program koji sračunava vrednosti niza logičkih formula zadatih u datoteci čije se ime prosledjuje sa standardnog ulaza. Formule su definisane nad azbukom

$$\Sigma = \{\&, |, =, >, < =, >, T, F, (,)\}$$

pri čemu se svi veznici zapisuju infiksno, sa uobičajenim prioritetom i asocijativnošću. Prilikom sračunavanja istinitosne vrednosti koristiti tehniku lenjog izračunvanja (*lazy evaluation*).

Prevodioci i interpretatori - Novembar 2003.

1. Proveriti da li su jezici $L_1 = (a^2 + b^*a^+)^*$ i $L_2 = ((a^2)^* + b^*a)^*$ jednaki. Odgovor obavezno dokazati.
2. U programskom jeziku C, tehnikom rekurzivnog spusta implementirati program koji sračunava vrednosti niza logičkih formula zadatih u datoteci čije se ime prosledjuje sa standardnog ulaza. Formule su definisane nad azbukom

$$\Sigma = \{\&, |, =, >, < =, >, T, F, (,)\}$$

pri čemu se svi veznici zapisuju infiksno, sa uobičajenim prioritetom i asocijativnošću. Prilikom sračunavanja istinitosne vrednosti koristiti tehniku lenjog izračunvanja (*lazy evaluation*).

Prevodioci i interpretatori - Novembar 2003.

1. Proveriti da li su jezici $L_1 = (a^2 + b^*a^+)^*$ i $L_2 = ((a^2)^* + b^*a)^*$ jednaki. Odgovor obavezno dokazati.
2. U programskom jeziku C, tehnikom rekurzivnog spusta implementirati program koji sračunava vrednosti niza logičkih formula zadatih u datoteci čije se ime prosledjuje sa standardnog ulaza. Formule su definisane nad azbukom

$$\Sigma = \{\&, |, =, >, < =, >, T, F, (,)\}$$

pri čemu se svi veznici zapisuju infiksno, sa uobičajenim prioritetom i asocijativnošću. Prilikom sračunavanja istinitosne vrednosti koristiti tehniku lenjog izračunvanja (*lazy evaluation*).

- 3 Data je sledeća podklasa C-datoteka: na početku se nalaze deklaracije globalnih promenljivih koje mogu biti samo osnovnog tipa (`int`, `char`, `float`), pri čemu eventualno mogu biti inicijalizovane odgovarajućim konstantama. Za deklaracijama sledi niz definicija funkcija osnovnog tipa, koje sadrže eventualno deklaracije lokalnih promenljivih i iskaze dodele oblika

```
prom = PozivFunkcije(...);
```

Napisati program (korišćenjem flex-a i yacc-a) koji proverava da li su promenljive inicijalizovane konstantama odgovarajućeg tipa i da li su ispravne dodele vrednosti funkcija promenljivama u okviru naredbi (funkcija treba da bude prethodno deklarirana u datoteci, kao i da bude istog tipa kao i promenqiva). U slučaju da neka od promenljivih nije deklarirana (ni kao lokalna ni kao globalna) javiti grešku.

- 3 Data je sledeća podklasa C-datoteka: na početku se nalaze deklaracije globalnih promenljivih koje mogu biti samo osnovnog tipa (`int`, `char`, `float`), pri čemu eventualno mogu biti inicijalizovane odgovarajućim konstantama. Za deklaracijama sledi niz definicija funkcija osnovnog tipa, koje sadrže eventualno deklaracije lokalnih promenljivih i iskaze dodele oblika

```
prom = PozivFunkcije(...);
```

Napisati program (korišćenjem flex-a i yacc-a) koji proverava da li su promenljive inicijalizovane konstantama odgovarajućeg tipa i da li su ispravne dodele vrednosti funkcija promenljivama u okviru naredbi (funkcija treba da bude prethodno deklarirana u datoteci, kao i da bude istog tipa kao i promenqiva). U slučaju da neka od promenljivih nije deklarirana (ni kao lokalna ni kao globalna) javiti grešku.

- 3 Data je sledeća podklasa C-datoteka: na početku se nalaze deklaracije globalnih promenljivih koje mogu biti samo osnovnog tipa (`int`, `char`, `float`), pri čemu eventualno mogu biti inicijalizovane odgovarajućim konstantama. Za deklaracijama sledi niz definicija funkcija osnovnog tipa, koje sadrže eventualno deklaracije lokalnih promenljivih i iskaze dodele oblika

```
prom = PozivFunkcije(...);
```

Napisati program (korišćenjem flex-a i yacc-a) koji proverava da li su promenljive inicijalizovane konstantama odgovarajućeg tipa i da li su ispravne dodele vrednosti funkcija promenljivama u okviru naredbi (funkcija treba da bude prethodno deklarirana u datoteci, kao i da bude istog tipa kao i promenqiva). U slučaju da neka od promenljivih nije deklarirana (ni kao lokalna ni kao globalna) javiti grešku.

- 3 Data je sledeća podklasa C-datoteka: na početku se nalaze deklaracije globalnih promenljivih koje mogu biti samo osnovnog tipa (`int`, `char`, `float`), pri čemu eventualno mogu biti inicijalizovane odgovarajućim konstantama. Za deklaracijama sledi niz definicija funkcija osnovnog tipa, koje sadrže eventualno deklaracije lokalnih promenljivih i iskaze dodele oblika

```
prom = PozivFunkcije(...);
```

Napisati program (korišćenjem flex-a i yacc-a) koji proverava da li su promenljive inicijalizovane konstantama odgovarajućeg tipa i da li su ispravne dodele vrednosti funkcija promenljivama u okviru naredbi (funkcija treba da bude prethodno deklarirana u datoteci, kao i da bude istog tipa kao i promenqiva). U slučaju da neka od promenljivih nije deklarirana (ni kao lokalna ni kao globalna) javiti grešku.