

Prevođenje programskih jezika - 2. praktični test [A]

Napomena: Na radnoj površini napraviti direktorijum sa nazivom oblika vašeg alas naloga. Npr. mi11056 za studenta sa indeksom 56/11. Neophodno je uz kod predati i Makefile i gramatiku sa skupovima izbora. Sintaksno neispravan kod se ocenjuje sa 0 od 10 poena.

1. Korišćenjem tehnike rekurzivnog spusta implementirati sintaksni analizator koji proverava ispravnost C-ovskog zaglavlja koje sadrži samo deklaracije funkcija. Zaglavlje mora da počne sa konkretna 2 reda, odnosno završi se sa konkretnim redom, oblika:

```
#ifndef _IME_ZAGLAVLJA_
#define _IME_ZAGLAVLJA_
.....
#endif
```

Izmedju, na mestu tačkica, nalaziće se samo deklaracije funkcija, ili nijedna.

Funkcija može da ima povratnu vrednost kada ispred naziva funkcije stoji konkretan tip, ili *void* ukoliko funkcija ne vraća vrednost. Posle naziva funkcije, u zagradama, navedeni su argumenti funkcije sa svojim tipovima, ili su zagrade prazne ako funkcija nema argumenta. Tipovi mogu biti samo standardni C-ovski tipovi. Naziv funkcije mora da počne slovom ili sa `_`, na mestima ostalih karaktera mogu se naći pored pomenutih karaktera i dekadne cifre. Isto pravilo važi i za argumente funkcija. Jednolinijski komentari koji počinju sa `//` se mogu naći unutar zaglavlja. Ispravni komentari ne utiču na sintaksnu ispravnost zaglavlja.

```
#ifndef Funkcije_h
#define Funkcije_h
    // deklaracije funkcija
    int f_ja_sa_argumentima(int a, float b, short* c);
    void f_ja_bez_argumenata();          // pomocna
    float _fja3 (char* s, int n);
    void _4_ta_funkcija_(float** M, int m_size, int n_size, double lambda);
    char*** poredi(char** a, char** b);
#endif
```

Prevođenje programskih jezika - 2. praktični test [B]

Napomena: Na radnoj površini napraviti direktorijum sa nazivom oblika vašeg alas naloga. Npr. mi11056 za studenta sa indeksom 56/11. Neophodno je uz kod predati i Makefile i gramatiku sa skupovima izbora. Sintaksno neispravan kod se ocenjuje sa 0 od 10 poena.

1. Korišćenjem tehnike simulacije rada potisnog automata implementirati sintaksni analizator koji proverava ispravnost C-ovskog zaglavlja koje sadrži samo deklaracije funkcija. Zaglavlje mora da počne sa konkretna 2 reda, odnosno završi se sa konkretnim redom, oblika:

```
#ifndef _IME_ZAGLAVLJA_
#define _IME_ZAGLAVLJA_
.....
#endif
```

Izmedju, na mestu tačkica, nalaziće se samo deklaracije funkcija, ili nijedna.

Funkcija može da ima povratnu vrednost kada ispred naziva funkcije stoji konkretan tip, ili *void* ukoliko funkcija ne vraća vrednost. Posle naziva funkcije, u zagradama, navedeni su argumenti funkcije sa svojim tipovima. Funkcija uvek mora imati bar jedan argument. Treba podržati da funkcije u C-u mogu imati i promenljiv broj argumenata kao npr. `printf` i njoj slične. Tada funkcija mora imati bar jedan konkretan argument koji je naveden na početku liste, a zatim se navode ... na kraju same liste argumenata. Tipovi argumenata mogu biti samo standardni C-ovski tipovi. Naziv funkcije mora da počne slovom ili sa `_`, na mestima ostalih karaktera mogu se naći pored pomenutih karaktera i dekadne cifre. Isto pravilo važi i za argumente funkcija. Komentari ogradjeni sa `/* */` se mogu naći unutar zaglavlja. Ispravni komentari ne utiču na sintaksnu ispravnost zaglavlja.

```
#ifndef Funkcije_h
#define Funkcije_h
    /* Deklaracije
       funkcija */
    int f_ja_sa_argumentima(int a, float b, short* c);
    void f_ja_bez_argumenata(int n, ...);    /*pomocna*/
    float _fja3 (char* s, int n);
    void _4_ta_funkcija_(float** M, int m_size, int n_size, double lambda, ... );
```