

## Prevođenje programskih jezika

Na Desktopu napraviti direktorijum sa nazivom **PPJ.sep1.ime.prezime.brojIndeksa.godina** i u njemu sačuvati svoj rad. Dakle, ako **Pera Perić** sa indeksom **123/2015** polaže ispit potrebno je da na Desktopu napravi direktorijum sa nazivom **PPJ.sep1.Pera.Peric.123.2015** i u njemu da sačuva svoj rad. **Sintaksno neispravni zadaci se ne pregledaju. Vreme za izradu ispita je 3h. Uslov za polaganje ispita je minimum 20p.**

### Sintaksna analiza navije

Nastavnici u osnovnim školama su nas zamolili da razvijemo jednostavan semantički kalkulator i tako pomognemo deci da savladaju operacije u skupu realnih brojeva. Semantički kalkulator podrazumeva da se iracionalni brojevi ne prikazuju svojim realnim aproksimacijama. Potrebno je da omogućimo sledeće:

1. Programski jezik treba da omogući jednostavno deklarisanje promenljivih navođenjem tipa podataka i imena promenljive. Opciono, moguće je dodeliti vrednost promenljivoj prilikom deklarisanja pomoću operatora **:=**. Program podržava racionalne i iracionalne brojeve. Racionalni brojevi se navode svojom decimalnom reprezentacijom, a iracionalni brojevi se definišu ključnom reči **sqrt** nakon čega se navodi pozitivan broj i opciono indeks korena. Omogućiti štampanje vrednosti jedne ili više promenljivih pomoću komande **stampaj**.

```
broj x;  
broj y := 10 + sqrt(2,3);  
broj z := -2.5 + sqrt(5);  
stampaj(x);  
stampaj(z, y);
```

0  
(10+sqrt(2,3)) (-2.5 + sqrt(5))

2. Programski jezik treba da omogući izračunavanja izraza. Podržane su osnovne aritmetičke operacije (**+**, **-**, **\***, **/**, **()**), izračunavanje suprotne vrednosti (**uminus**) i stepenovanje pomoću operatora **^**. Ukoliko se prilikom deljenja dobije iracionalni broj u imeniocu, konačni rezultat treba racionalisati. Izrazi mogu biti samostalne naredbe i u tom slučaju se odmah štampa vrednost izraza. Takođe, treba omogućiti izmenu vrednosti promenljive.

```
x := 2*z + y;  
x + 3;  
stampaj(-z*2);  
(z + sqrt(5))^2;
```

8 + 2\*sqrt(5) + sqrt(2,3)  
5 - 2\*sqrt(5)  
26.25 + 10\*sqrt(5)

3. Programski jezik treba da omogući upoređivanje vrednosti izraza (**<**, **<=**, **>**, **>=**, **==**, **!=**).

```
x > y;  
x+(y - 3*z)^4 <= -y / (z + 2*y)^2;
```

True  
True

4. Proširiti programski jezik tako da podržava rad sa nizovima realnih brojeva. Niz se definiše ključnom reči **niz** nakon čega treba navesti naziv niza i listu vrednosti koje sadrži. Nizovi se prvo moraju definisati, pre nego što se koriste. Potrebno je modifikovati funkciju **stampaj** tako da podržava štampanje nizova :

```
niz A := [x, 2*y, 2 + sqrt(3)];  
stampaj(A);
```

[2+2\*sqrt(5)+sqrt(2,3), 20+2\*sqrt(2,3) ... ]

5. Proširiti programski jezik tako da podržava primenu aritmetičkih operacija nad nizovima. Ukoliko se aritmetičke operacije primenjuju nad dva niza, tada se rezultat dobija tako što se operacije primene pokoodinarno. Omogućiti indeksiranje niza i izmenu vrednosti elemenata u nizovima.

```
A := [2 + 5*sqrt(6), x];  
B := sqrt(6) * A;  
stampaj(B[0]);
```

30 + 2\*sqrt(6)

**Napomena:** *Makefile* je obavezen deo rešenja.

## Sintaksna analiza naniže

Uz pomoć **potisnog automata** implementirati sintaksni analizator koji omogućava proveru sintaksne ispravnosti dela pod 1 iz prethodnog zadatka. Podrazumevati da prilikom definicije promenljive nije moguće koristiti izraze. U komentaru programa je neophodno ostaviti sređenu gramatiku i skupove izbora.

```
broj x;  
broj y := 10 + sqrt(2,3);  
broj z := -2.5 - sqrt(5);  
stampaj(x);  
stampaj(z, y);
```

**Ispit traje 3h.**