

Prevođenje programskih jezika

Na Desktopu napraviti direktorijum sa nazivom **PPJ.jan1.ime.prezime.brojIndeksa.godinaUpisa** 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.jan1.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 naviše

Potrebno je napisati interpreter za zamišljeni programski jezik koji podržava bitovske operacije nad celim brojevima:

1. Interpreter treba da omogući jednostavno definisanje promenljivih nalik na C i štampanje njihovih vrednosti u izabranoj osnovi (2, 10, 16). Operatori konverzije za osnove su redom %b, %d i %h.

```
def x;                // definiše promenljivu x čija je vrednost 0
def y = 10;           // definiše promenljivu y čija je vrednost 10
print("%d", x);       // na stdout štampa vrednost promenljive x u osnovi 10
print("%b)y");        // na stdout štampa vrednost promenljive y u osnovi 2
```

2. Interpreter treba da omogući sve uobičajene aritmetičke operacije nad promenljivama i konstantama (+, -, *, /, %, (), unarni minus). Proširiti operacije definisanja promenljivih i štampanja tako da podržavaju aritmetičke izraze.

```
x = 4 + y % 2;
def z = -3*y + x/7;;
print("%h", 4*z - 7*(5-3)); // na stdout štampa vrednost izraza u osnovi 16
```

3. Interpreter treba da omogući sve uobičajene bitovske operacije nad izrazima. (&, |, ^, ~, <<, >>). Proširiti operacije definisanja promenljivih i štampanja tako da podržavaju i bitovske izraze.

```
def w = (~x<<3)|z + (y>>2)&(1<<5);
print("%d", w^x); // na stdout štampa vrednost izraza u osnovi 10
```

4. Interpreter treba da omogući sve uobičajene logičke operacije nad izrazima (<, <=, >, >=, ==, !=).

```
x + y == -y&x; // na stdout štampa True/False u zavisnosti od uslova
```

5. Modifikovati interpreter tako da rezultat interpretacije bude reprezentacija programa pomoću sintaksnog stabla. Sintaksno stablo treba da omogući interpretaciju, tj. izvršavanje programa i štampanje stabla na standardni izlaz.

Napomena: Makefile je obavazan deo rešenja.

Napomena 2: Zadaci 1-4 i zadatak 5 se odvojeno pregledaju. Zbog toga, rešenja zadataka 1-4 i zadatka 5 sačuvajte u odvojenim direktorijumima unutar svog direktorijuma. Na primer, rešenja zadataka 1-4 sačuvajte u direktorijumu „Interaktivni parser“, a rešenje zadatka 5 sačuvajte u direktorijumu „Sintaksno stablo“.