

## Prevođenje programskih jezika - Septembar 2015.

1. Napisati program koji omogućava korisniku rad sa iskaznim formulama.

- (a) Iskazne formule se dobijaju rekurzivno od logičkih promenljivih (čija imena se sastoje od proizvoljnog broja malih slova engleske abecede i cifara), logičkih konstanti **True** i **False**, konačnom primenom konjunkcije, disjunkcije, implikacije, ekvivalencije i negacije nad već definisanim formulama. Formule mogu da sadrže i zagrade. Napraviti interpreter koji proverava sintaksnu ispravnost formula koje se unose sa ulaza.

```
p /\ q <=> r
!p => q \/ r <=> s
p1 /\ (p2 <=> !p3)
p /\ /\ r                                     syntax error
```

- (b) Interpreteru napravljenom u delu pod a, dodati naredbu dodele:

```
p := True
q := False
r := p <=> q
```

- (c) Unaprediti interpreter tako da računa vrednost zadate formule:

```
p := True
q := False
r := p <=> q
p /\ q <=> r <=> True           True
!p => q \/ r <=> s              Variable s is not defined.
```

- (d) Omogućiti da u slučaju korišćenja promenljive koja dotada nije definisana, program ipak sračuna vrednost formule, na taj način što će dati konkretnu vrednost (**True** ili **False**) ako vrednost formule ne zavisi od neinicijalizovane promenljive, a inače je vrednost formule nova logička konstanta **Undef**.

```
p := True
q := False
p /\ q \/ s           Undef
p \/ s \/ q           True
!(q => s)              False
```

- (e) Interpreter podržava rad sa listama, nad kojima je dozvoljen indeksni operator pristupa, kao i operatori *head* i *tail* koji vraćaju prvi element liste, odnosno rep liste.

```
p := True
L := [p, !p, p <=> p, False]
L[2] \/ L[1]           True
head(L) /\ head (tail (tail (tail (L)))) False
```