Kompilacija programskih jezika - Jun 2013.

praktični deo

- 1. Napisati interpreter za rad sa regularnim izrazima nad azbukom $\Sigma = \{a, ..., z\}$:
 - (a) Prepoznati validne regularne izraze: (tačka predstavalja konkatenaciju)

```
a*.b+.c.d.e{3}|a+.b?.c.d|[abcde]|(a.b)*.[cde]?
```

(b) Regularni izrazi se naredbom BASIC prebacuju u osnovne regularne izraze:

(c) Omogućiti rad sa regularnim definicijama:

```
DEFINITION A = a.b*|b.a* , B = [ab]{2} 
BASIC A  (((a)((b)*))|((b)((a)*)))
```

(d) Regularne definicije se prave nad već definisanim regularnim definicijama ili nad regularnim izrazima:

(e) Za zadati regularni izraz naći dužinu najduže reči koja je opisana njime:

```
MAXLEN a*.b{3}.[abc]?. C +Inf MAXLEN a?.[abc].b{3}|abc 5
```

Srećno!

Kompilacija programskih jezika - Jun 2013.

praktični deo

- 1. Napisati interpreter za rad sa regularnim izrazima nad azbukom $\Sigma = \{a, ..., z\}$:
 - (a) Prepoznati validne regularne izraze: (tačka predstavalja konkatenaciju)

```
a*.b+.c.d.e{3}|a+.b?.c.d|[abcde]|(a.b)*.[cde]?
```

(b) Regularni izrazi se naredbom BASIC prebacuju u osnovne regularne izraze:

(c) Omogućiti rad sa regularnim definicijama:

```
DEFINITION A = a.b*|b.a* , B = [ab]{2} 
BASIC A  (((a)((b)*))|((b)((a)*)))
```

(d) Regularne definicije se prave nad već definisanim regularnim definicijama ili nad regularnim izrazima:

(e) Za zadati regularni izraz naći dužinu najduže reči koja je opisana njime:

```
MAXLEN a*.b{3}.[abc]? . C +Inf MAXLEN a?.[abc].b{3}|abc 5
```