Μεταγλωτιστές 2019 Προγραμματιστική Εργασία #2

Ονοματεπώνυμο: Διονύσιος Κοντοζίδης

АМ: П2016178

Κανόνες Γραμματικής

```
Grammar
Stmt list →
             Stmt Stmt list
Stmt →
             id equal Exp
            | print Exp.
             Term Term tail.
Exp →
Term tail → xor Term Term tail
             Factor Factor tail.
Factor tail →or Factor Factor tail
Factor →
             Atom Atom tail.
Atom tail →
             and Atom Atom tail
Atom →
             lparen Exp rparen
            | id
            | number.
```

1.1 Σύνολο κανόνων γραμματικής

Στην παραπάνω εικόνα είναι το σετ κανόνων γραμματικής που χρησιμοποιήθηκε για την υλοποίηση της εργασίας. Στην διαδικασία αυτή χρησιμοποιήθηκε η μέθοδος της αναδρομικής κατάβασης. Η γραμματική κατασκευάστηκε με τέτοιο τρόπο ώστε να δέχεται συμβολοσειρές με bits και λογικές πράξεις μεταξύ αυτών με προτεραιότητα:

- and
- or
- xor

Εξαιτίας αυτού στην γραμματική τοποθετήθηκε το and στο **term_tail**,το or στο **factor_tail** και το xor στο **atom_tail** γιατί όσο πιο κάτω είναι ένας τελεστής στους κανόνες γραμματικής, τόσο πιο μεγάλη προτεραιότητα έχει.

Αποτελέσματα Ελέγχου για LL(1) συμβατότητα

- All nonterminals are reachable and realizable.
- The nullable nonterminals are: Stmt_list Term_tail Factor_tail Atom_tail.
- The endable nonterminals are: Atom_tail Atom Factor_tail Factor Term_tail Term Exp Stmt_list Stmt.
- · No cycles.

nonterminal	first set	follow set	nullable	endable
Stmt_list	id print	Ø	yes	yes
Stmt	id print	id print	no	yes
Term_tail	xor	rparen id print	yes	yes
Term	lparen id number	rparen xor id print	no	yes
Factor_tail	or	rparen xor id print	yes	yes
Factor	lparen id number	rparen or xor id print	no	yes
Atom_tail	and	rparen or xor id print	yes	yes
Atom	lparen id number	rparen and or xor id print	no	yes
Exp	lparen id number	rparen id print	no	yes

The grammar is LL(1).

2.1 Αποτελέσματα για LL(1) συμβατότητα

Η γραμματική είναι LL(1) όπως φαίνεται και παραπάνω. Αυτό σημαίνει ότι το διάβασμα των κανόνων γίνεται από τα αριστερά στα δεξιά και ότι διαθέτετε μονοπάτι πρός όλα τα μη τερματικά σύμβολα. Ακόμα, δεν υπάρχουν κυκλικές μεταβάσεις και συνεπώς συμφωνεί με την μέθοδο της αναδρομικής κατάβασης που προαναφέρθηκε.

Πίνακας με FIRST και FOLLOW σετ της γραμματικής

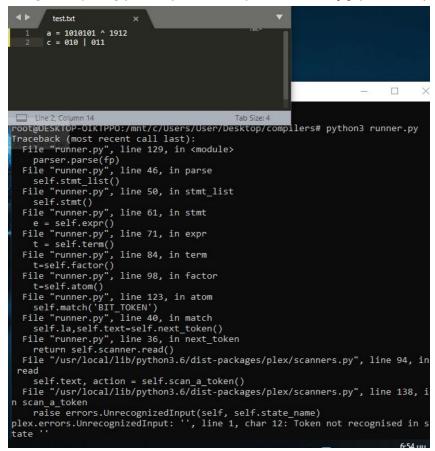
Μη τερματικά	FIRST	FOLLOW
Stmt_list	id print	3
Stmt	id print	id print
Term_tail	xor) id print ε
Term	(id number) xor id print
Factor_tail	or) xor id print ε
Factor	(id number) or xor id print
Atom_tail	and) or xor id print ε
Atom	(id number) and or xor id print
Exp	(id number) id print

Τα FIRST και FOLLOW σετ χρησιμοποιούνται ώστε ο συντακτικός αναλυτής να είναι σε θέση να εφαρμόσει τον σωστό κανόνα στην σωστή θέση.

Αποτελέσματα Εξόδου

```
a = 1001
        b = 101 xor 111
        c = 1001100 and 0001001
        print (10101 and 10111 or 11001 xor 00001)
        print a
        print 1 or 0 xor 1
        print c
        print b
 Line 8, Column 8
                                                Tab Size: 4
root@DESKTOP-OIKTPPO: /mnt/c/Users/User/Desktop/compilers
root@DESKTOP-OIKTPPO:/mnt/c/Users/User/Desktop/compil
11100
1001
1000
root@DESKTOP-OIKTPPO:/mnt/c/Users/User/Desktop/compil
```

4.1 Είσοδος και έξοδος μετά την εκτέλεση του runner.py με σωστή είσοδο



4.2 Είσοδος και έξοδος μετά την εκτέλεση του προγράμματος **runner.py** με λάθος είσοδο

<u>Πηγές</u>

Χρησιμοποιήθηκε το Context Free Grammar Tool για τον έλεγχο της συμβατότητας της γραμματικής και τον υπολογισμό των FIRST και FOLLOW set.

Link: http://smlweb.cpsc.ucalgary.ca/start.html