

Μεταγλωττιστές 2018 Προγραμματιστική Εργασία #2

Ονοματεπώνυμο: Αστέριος Παπαμιχαήλ
ΑΜ: Π2015059

A. Η γραμματική η οποία βγήκε από τα ζητούμενα είναι η εξής:

```
Stmt_list  -> Stmt Stmt_list | ε
Stmt       -> id assign Expr | print Expr
Expr       -> Term Term_tail
Term_tail  -> OrOp Term Term_tail | ε
Term       -> Notop Factor Factor_tail
Factor_tail -> AndOp Notop Factor Factor_tail | ε
Factor     -> ( Expr ) | id | BoolValue
BoolValue  -> true_token | false_token
OrOp       -> or
AndOp      -> and
Notop      -> not | ε
```

B1. Το αποτέλεσμα για LL(1) είναι το ακόλουθο:

The grammar is LL(1).

- All nonterminals are reachable and realizable.
- The nullable nonterminals are: Stmt_list Term_tail Factor_tail Notop.
- The endable nonterminals are: BoolValue Factor_tail Factor Term_tail Term Expr Stmt_list Stmt.
- No cycles.

Some sentences generated by this grammar: {ε, print id, print not id, id assign id, print true_token, id assign not id, print false_token, id assign id and id, print not id and id, print id and not id, print not true_token, id assign true_token, print not false_token, id assign false_token, id assign not id and id, id assign id and not id, print not id and not id, id assign not true_token, id assign not false_token, id assign not id and not id}

B2. Ο πίνακας με τα first και follow sets της παραπάνω γραμματικής για όλα τα μη τερματικά σύμβολα:

FIRST-set	Μη τερματικά σύμβολα	FOLLOW-set
id, print	Stmt_list	-
id, print	Stmt	id, print
or	Term_tail), id, print
not, (, id, true_token, false_token	Term), or, id, print
and	Factor_tail), or, id, print
(, id, true_token, false_token	Factor), and, or, id, print
not, (, id, true_token, false_token	Expr), id, print
true_token, false_token	BoolValue), and, or, id, print
or	OrOp	not, (, id, true_token, false_token
and	AndOp	not, (, id, true_token, false_token
not	Notop	(, id, true_token, false_token

Γ. Περιγραφή των αρχείων.

Αρχικά το αρχείο parser.py κάνει λεκτική και συντακτική ανάλυση της εισόδου χρησιμοποιώντας αναδρομικές κλήσεις συναρτήσεων της γλώσσας python και επιστρεφόμενες τιμές. Όπου κάθε όνομα συνάρτησης αντιστοιχεί και σε ένα όνομα μη τερματικού συμβόλου της γραμματικής. Στην κλάση του συντακτικού αναλύτη αποθηκεύεται και η θέση όπου είναι ο "κερσοράς" του λεξικού αναλύτη όπου σε περίπτωση σφάλματος να εμφανίζει σε ποία γραμμή και σε ποίο κομάτι της γραμής προέκυψε το σφάλμα.

Και το αρχείο runner.py ουσιαστικά χρησιμοποιώντας τον κώδικα του προαναφερόμενου αρχείου και την δομή του λεξικού (dictionary) της python όπου το χρησιμοποιεί για την αποθήκευση και προσπέλαση των μεταβλητών που ορίζονται από τον χρήστη της ζητούμενης γραμματικής, εκτελεί πλέον το αρχείο του χρήστη.

Δ. Αποτελέσματα εξόδου

```

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ cat test.txt
y = f
x = t and not (True and not (False or y))
print x

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 parser.py
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 runner.py
False

```

```
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ cat test.txt
x1 = 1
x2 = 0
print x1 and not x2

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 parser.py
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 runner.py
True
```

```

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ cat test.txt
y = f
x = t and not (True and not (False or y))
print x

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 parser.py
ParseError:
  missing ) at line 1 :
  x = t and not (True and not (False or y))
                                   ^

steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 runner.py
ParseError:
  missing ) at line 1 :
  x = t and not (True and not (False or y))
                                   ^

steve@steve-vBox:/media/sf_sharedFolderU/compilers$

```

[illegible]

```
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ cat test.txt
print t and f or not f
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 parser.py
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 runner.py
True
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ █
```

```
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ cat test.txt
x0 = 1
y0 = 0
x1 = T
y1 = false

print y0 or not ( x0 and not( x1 ) or y1)
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 parser.py
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ python3 runner.py
True
steve@steve-vBox:/media/sf_sharedFolderU/compilers$ █
```