

Μεταγλωττιστές 2018

Προγραμματιστική Εργασία

Ονοματεπώνυμο: Τουλάκης Αλέξανδρος

AM: Π2015067

Κανόνες της γραμματικής

Stmt_list -> Stmt Stmt_list | ε

Stmt -> id assign Expr | print Expr

Expr -> Term Term_tail

Term_tail -> Or_op Term Term_tail | ε

Term -> Negative_factor Factor_tail

Factor_tail -> And_op Negative_factor Factor_tail | ε

Negative_factor -> Not_op Factor

Factor -> (Expr) | id | Boolean

Boolean -> true | false

Or_op -> or

And_op -> and

Not_op -> not | ε

Γραμματική LL(1) First and Follow Sets

nonterminal	first set	follow set	nullable	endable
Stmt_list	id print	∅	yes	yes
Stmt	id print	id print	no	yes
Expr	not (Expr) id true false	id print	no	yes
Term_tail	or	id print	yes	yes
Term	not (Expr) id true false	or id print	no	yes
Factor_tail	and	or id print	yes	yes
Negative_factor	not (Expr) id true false	and or id print	no	yes
Factor	(Expr) id true false	and or id print	no	yes
Boolean	true false	and or id print	no	yes
Or_op	or	not (Expr) id true false	no	no
And_op	and	not (Expr) id true false	no	no
Not_op	not	(Expr) id true false	yes	no

The grammar is LL(1).

Εξήγηση του κώδικα

Το πρόγραμμα αναγνώρισης της γλώσσας (parser.py) αρχικά δημιουργήθηκε ένα λεξικό στην plex το οποίο περιλαμβάνει όλα τα πιθανά TOKENS τα οποία μπορεί να υποστηρίξει το πρόγραμμα. Για να προσδιοριστεί η ορθότητα της εισόδου του χρήστη γίνεται χρήση της μεθόδου της αναδρομικής κατάβασης. Η μέθοδος αυτή καλεί αναδρομικά συναρτήσεις, οι οποίες αναπαριστούν μη τερματικές καταστάσεις μέχρι να φτάσει σε μια βασική περίπτωση. Οι βασικές περιπτώσεις στο πρόγραμμα μας βρίσκονται μέσα στην συνάρτηση Factor. Λάθος θα προκύψει στην περίπτωση που ο χρήστης δώσει μια είσοδο για την οποία το πρόγραμμα δεν θα μπορέσει να βρει διαθέσιμη επιλογή.

Το πρόγραμμα (runner.py) έχει παρόμοια λειτουργικότητα με αυτήν του parser.py καθώς για να φτάσει στο σημείο να εκτελέσει τον κώδικα που έχει δώσει ο χρήστης θα πρέπει πρώτα να διαπιστώσει την ορθότητα του. Για την εκτέλεση εντολών ανάθεσης σε μεταβλητή υπάρχει ένα λεξικό στο οποίο ως κλειδί έχουμε το όνομα της μεταβλητής και ως τιμή αλήθεια ή ψέμα. Για την εμφάνιση στην οθόνη ενός αποτελέσματος χρησιμοποιείται η εγγενής συνάρτηση print της python σε ένα expression. Τέλος για τους τελεστές που έχουν δύο μέρη ο κώδικας δημιουργεί αναδρομικά και τα δύο μέρη των τελεστών και τρέχει τις κατάλληλες πράξεις(and , or).

Αποτελέσματα Εκτέλεσης

Επειδή σε περίπτωση λάθους ο runner και ο parser θα βγάλουν το ίδιο μήνυμα σφάλματος και σε σωστή είσοδο ο runner θα τρέξει τα αποτελέσματα και ενδεχομένως να εμφανίσει αποτελέσματα ενώ ο parser όχι οι εικόνες προέρχονται από τον runner.py

Λάθος αποτελέσματα

```
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>more data.txt
print 1 or true or not(true and f
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>python runner.py
Parser Error: found None instead of ) at line 1 char 34
```

```
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>more data.txt
print x
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>python runner.py
Parser Error: Variable x is not initialised ! at line 1 char 8
```

```
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>more data.txt
print not(and or True or false and not (True and false))
print x
print print and false
false and false or false
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>python runner.py
factor - (
Parser Error: Expected : ( ID TRUE FALSE not at line 1 char 11
```

Σωστά Αποτελέσματα

```
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>more data.txt
print 1 or true or not(true and f)
x = 1 and f
print x and false
z = x or not x
print z
print 1 or true or not(true or not(true))
C:\Users\AlexToul\Desktop\Σχολή\ΣΤ Εξάμηνο\compilers\final\final>python runner.py
True
False
True
True
```

Δικτυογραφία

- 1) <http://mixstef.github.io/courses/compilers/lecturedoc/unit4/module1.html> (plex)
- 2) https://en.wikipedia.org/wiki/LL_parser
- 3) <https://www.youtube.com/watch?v=9vmhcBpZDcE>
- 4) https://en.wikipedia.org/wiki/LL_grammar