

Μεταγλωττιστές 2018

Προγραμματιστική Εργασία

Ονοματεπώνυμο: Τουλάκης Αλέξανδρος

ΑΜ: Π2015067

Κανόνες της γραμματικής

Stmt_list -> Stmt Stmt_list | .

Stmt -> id assign Expr | print Expr.

Expr -> Term Term_tail.

Term_tail -> Or_op Term Term_tail | .

Term -> Negative_factor Factor_tail.

Factor_tail -> And_op Negative_factor Factor_tail | .

Negative_factor -> Not_op Factor.

Factor -> (Expr) | id | Boolean.

Boolean -> true | false.

Or_op -> or.

And_op -> and.

Not_op -> not | .

Γραμματική LL(1) First and Follow Sets

nonterminal	first set	follow set	nullable	endable
Stmt_list	id print	\emptyset	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

Και τα δύο αρχεία δημιουργούν ένα λεξικό το οποίο περιλαμβάνει όλα τα υποστηριζόμενα TOKENS τα οποία μπορεί να αναγνωρίσει ο Parser και ο Runner. Το Πρόγραμμα του Parser

Προσπαθεί να διαπιστώσει οι εντολές μας είναι σωστές το άμα είναι σωστές προσδιορίζονται από την γραμματική για αυτό τον λόγο χρησιμοποιείται η τεχνική της αναδρομικής κατάβασης.

Η λειτουργικότητα του Parser και του runner έχει παρόμοια στην λογική στο σημείο ότι

Και τα δύο προγράμματα δημιουργούν λεξικό με τα υποστηριζόμενα TOKENS της γλώσσας στο PLEX . Στην συνέχεια προσπαθεί να αναγνωρίσει άμα η γραμματική είναι σωστή . Για να το πετύχει αυτό καλεί αναδρομικά συναρτήσεις μέχρι να φτάσει σε μία βασική περίπτωση στην περίπτωση μας είναι το Factor το οποίο αναγνωρίζει (Expr) η κάποιο identifier η κάποια σταθερά τύπου Boolean. Τα αποτελέσματα που βγάζουν επιστρέφονται στις συναρτήσεις που τις καλούν και έτσι μπορούν να εκτελεστούν λειτουργίες. Σε περίπτωση που μία επιλογή δεν οδηγεί σε σωστά αποτελέσματα δοκιμάζουμε τις αμέσως επόμενες πιθανά σωστές

γραμματικές και άμα καμία δεν βρεθεί τότε ρίχνεται exception το οποίο ενημερώνει τον χρήστη για πιθανά λάθη που έκανε.

Στον runner εφόσον η γραμματική μας είναι σωστή τότε θα εκτελεστεί κάποια ενέργεια π.χ

Θα εκτυπώσει κάποιο μήνυμα στην οθόνη για το αποτέλεσμα τις λογικής έκφρασης. Παρόλα αυτά επειδή δεν ξέρουμε πάντα άμα μία έκφρασή θα είναι ολοκληρωτικά σωστή π.χ

Στην περίπτωση του Factor Tail στο σημείο που είμαστε μπορεί να τρέξει κανονικά ενώ όμως στα επόμενα TOKENS το πρόγραμμα όμως θα εκτελεί τις πράξεις μέχρι να φτάσει στο τέλος.

Αποτελέσματα Εκτέλεσης

Λάθος αποτελέσματα

- 1) print not(and or True or false and not (True and false))
- 2) print x
- 3) print print and false
- 4) false and false or false

```
C:\Users\AlexToul\Desktop\stef>python runner.py
perrExpected: '(' or identifier or not or boolean value
C:\Users\AlexToul\Desktop\stef>
```

```
C:\Users\AlexToul\Desktop\stef>python runner.py
Traceback (most recent call last):
  File "runner.py", line 205, in <module>
    p.parse(fp)
  File "runner.py", line 82, in parse
    self.stmt_list()
  File "runner.py", line 86, in stmt_list
    self.stmt()
  File "runner.py", line 101, in stmt
    print(self.expr())
  File "runner.py", line 107, in expr
    t = self.term()
  File "runner.py", line 136, in term
    f = self.negative_factor()
  File "runner.py", line 166, in negative_factor
    return revbool_value(self.factor(), negative)
  File "runner.py", line 179, in factor
    raise RuntimeError("Unitialized variable: ", varname)
__main__.RuntimeError: ('Unitialized variable: ', 'x')
```

```
C:\Users\AlexTou1\Desktop\stef>python runner.py
perrExpected: '(' or identifier or not or boolean value
C:\Users\AlexTou1\Desktop\stef>
```

```
C:\Users\AlexTou1\Desktop\stef>python runner.py
perrExpected: identifier or print
C:\Users\AlexTou1\Desktop\stef>
```

Σωστά Αποτελέσματα

- 1) id = True and true or true
print id
- 2) print (true or false and true or FALSE)
- 3) print (true or false and true or not FALSE and (true or 1 or t))
- 4) print (f and t and not TRUe)

1-3 είναι True

```
C:\Users\AlexTou1\Desktop\stef>python runner.py
True
C:\Users\AlexTou1\Desktop\stef>
```

4 είναι False

```
C:\Users\AlexTou1\Desktop\stef>python runner.py
False
C:\Users\AlexTou1\Desktop\stef>
```

Δικτυογραφία

- 1) <http://mixstef.github.io/courses/compilers/lecturedoc/unit4/module1.html> (plex)
- 2) https://en.wikipedia.org/wiki/LL_parser
- 3) https://en.wikipedia.org/wiki/LL_grammar

