- **Κανόνες της Γραμματικής:**

```
Stmt_list -> Stmt Stmt_list
          |.
Stmt -> id = Expr
     | print Expr.
Expr -> Term Term_tail.
Term_tail -> Aoop Term Term_tail
          |.
Term -> Factor Factor_tail.
Factor_tail -> Nop Factor Factor_tail
            |.
Factor -> (Expr)
       | id
       | tf01.

Aoop -> and
        | or.
Nop -> not.
```

- **Από το παρακάτω στιγμιότυπο και με τη βοήθεια του online εργαλείου http://smlweb.cpsc.ucalgary.ca/start.html, επιβεβαιώνεται ότι η γραμματική που σχεδιάστηκε είναι LL(1).**

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

nonterminal	first set	follow set	nullable	endable
Stmt_list	id print	∅	yes	yes
Stmt	id print	id print	no	yes
Expr	(Expr) id tf01	id print	no	yes
Term_tail	and or	id print	yes	yes
Term	(Expr) id tf01	and or id print	no	yes
Factor_tail	not	and or id print	yes	yes
Factor	(Expr) id tf01	not and or id print	no	yes
Aoop	and or	(Expr) id tf01	no	no
Nop	not	(Expr) id tf01	no	no

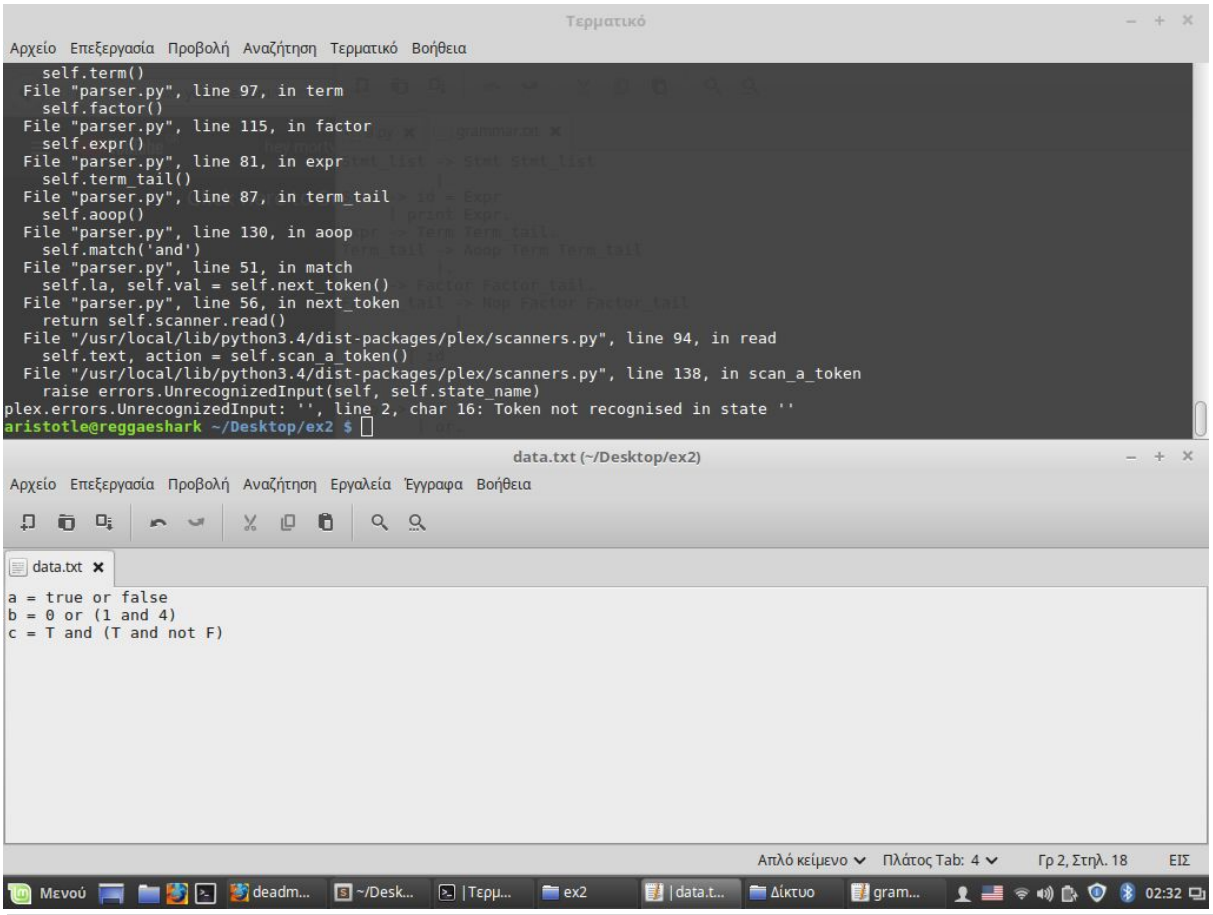
The grammar is LL(1).

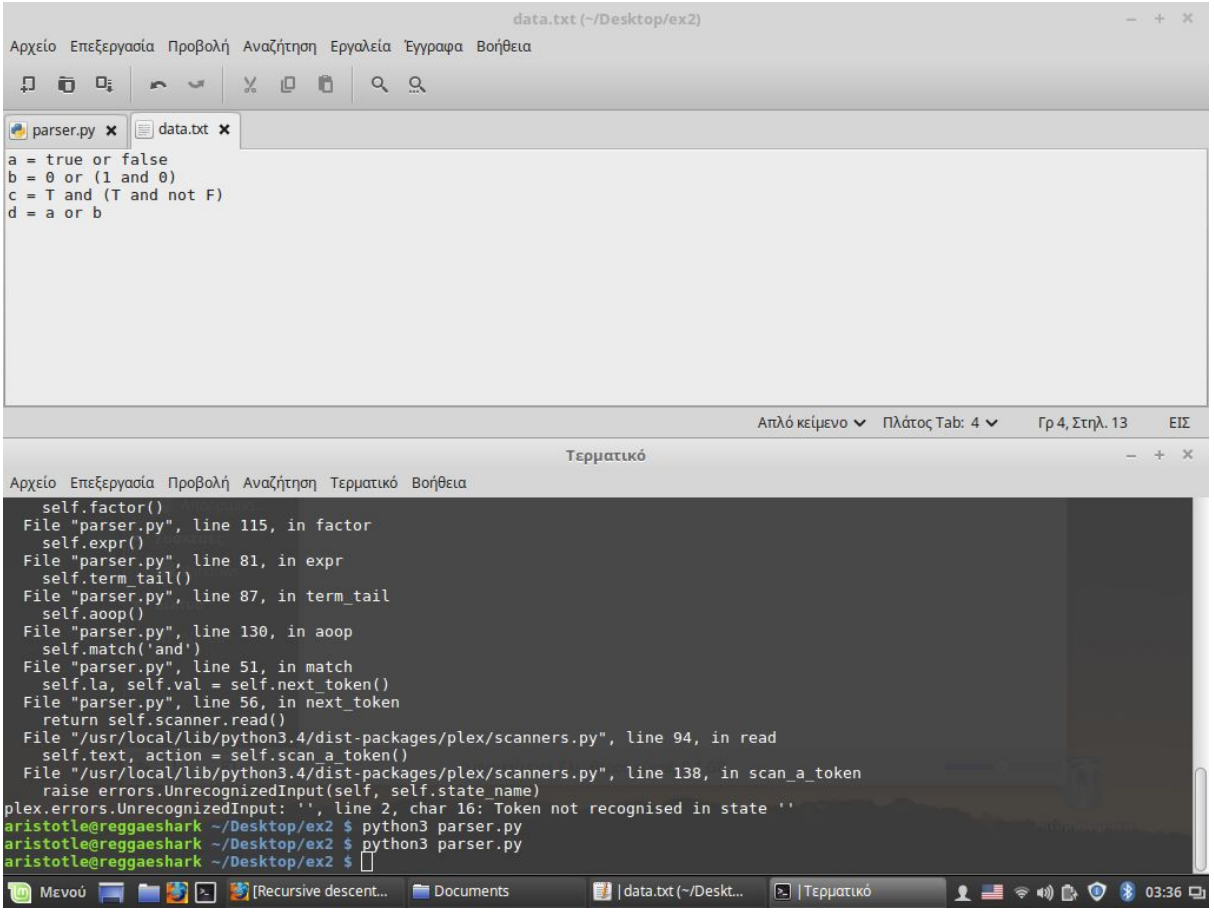
- Επιπλέον, είναι διακριτά και τα FIRST / FOLLOW sets της:
    - FIRST sets:
        - Stmt_list: id =
        - Stmt: id = print
        - Expr: (Expr) id tf01
        - Term_tail: and or
        - Term: (Expr) id tf01
        - Factor_tail: not
        - Factor: (Expr) id tf01
        - Aoop: and or
        - Nop: not
    - FOLLOW sets:
        - Stmt_list
        - Stmt: id print
        - Expr: id print
        - Term_tail: id print
        - Term: and or id print
        - Factor_tail: and or id print
        - Factor: not and or id print
        - Aoop: (Expr) id tf01
        - Nop: (Expr) id tf01

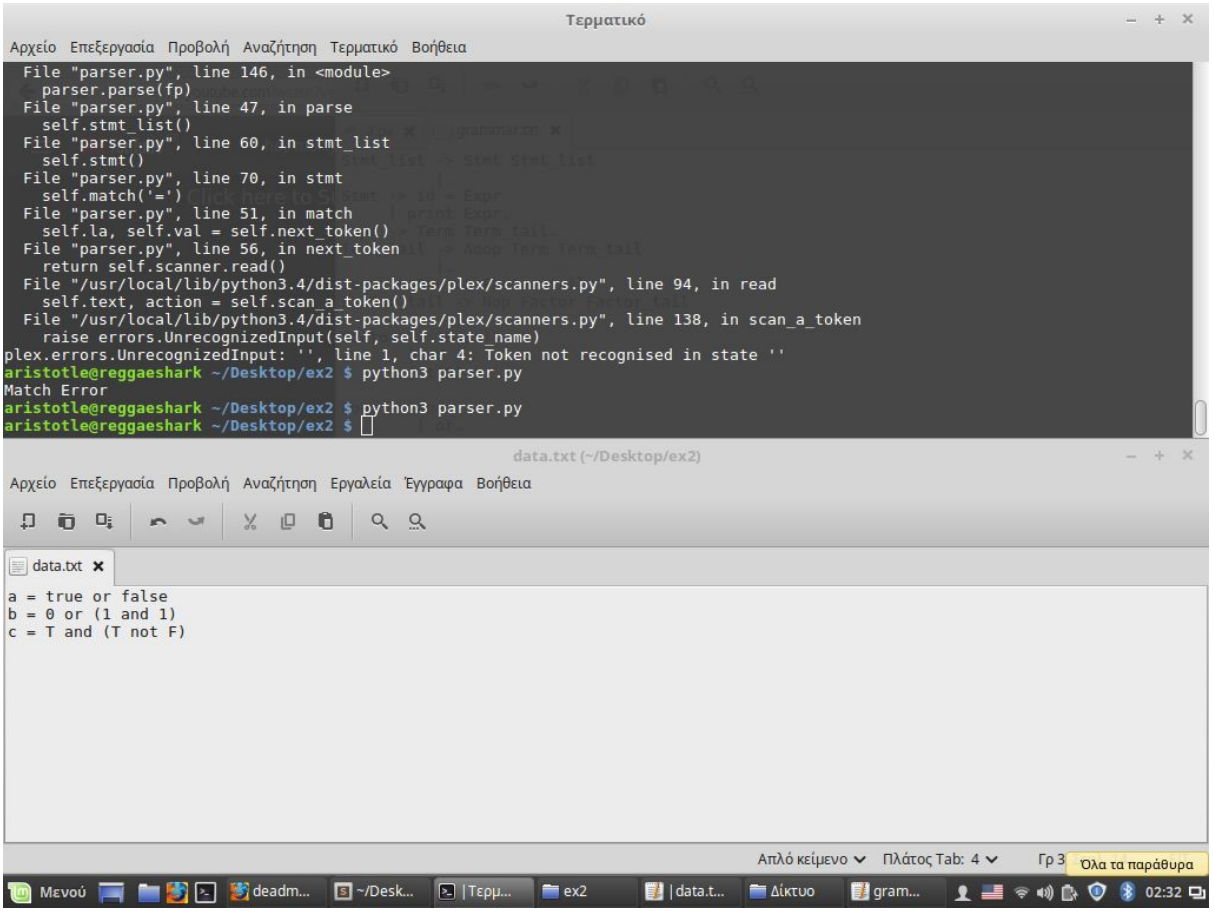
- **<u>Συνοπτική Περιγραφή του κώδικα *parser.py:*</u>**

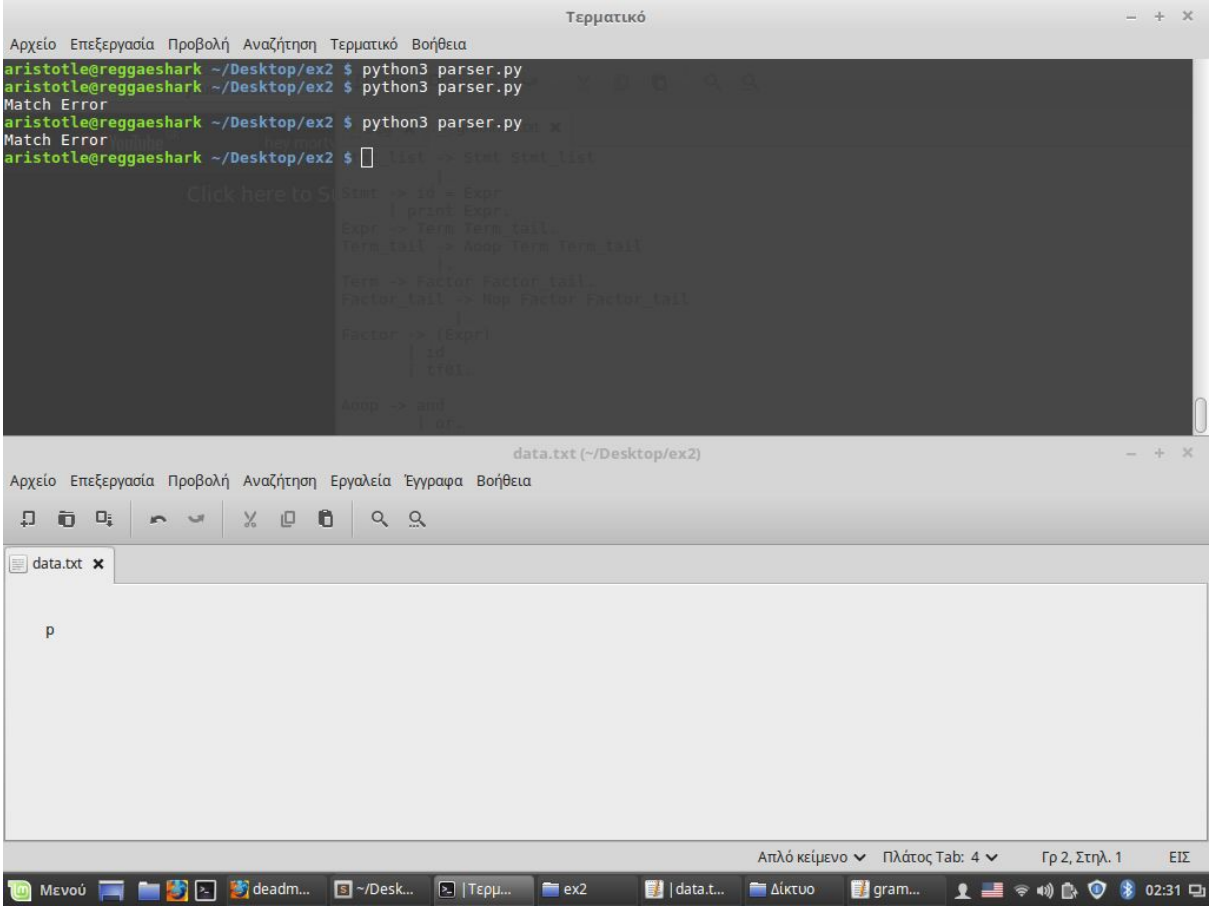
Αρχικά, ο κώδικας ξεκινάει, δημιουργώντας το scanner, και έπειτα, δημιουργούμε το λεξικό, στο οποίο θα βασιστεί το πρόγραμμα. Βάζω τους χαρακτήρες *t, f, 0, 1* και τις λέξεις *true, false.* Επιπλέον, χρησιμοποιούνται και οι τελεστές *not, and, or.* Ξεκινώντας, μέσω της συνάρτησης *parse,* ξεκινάει δημιουργείται ο *scanner* και προχωράει στην συνάρτηση *stmt_list.* Έπειτα, ψάχνει κάποιο *Identifier*, δηλαδή το όνομα μίας μεταβλητής, που το δίνουμε στο αρχείο που διαβάζει. Εφόσον διαβάσει το *identifier* ή τον χαρακτήρα *'='*, συνεχίζει είτε στην *stmt_list* ή προχωράει στην *stmt*. Από την *stmt,* διαβάζει από το *stmt_list,* το όνομα κάποιας μεταβλητής, = ή προχωράει και διαβάζει την εντολή *print.* Μετά, προχωραει στην συνάρτηση *expr,* στη συνέχεια στην συνάρτηση, *term(),* η οποία συνεχίζει στην συνάρτηση *factor(),* η οποία ανάλογα με το input, διαβάζει είτε ένα expression μέσα από παρενθέσεις, ή κάποιο identifier, η κάποιο από τα (*true, t, 1), (false, f, 0).* Το πρόγραμμα συνεχίζει έτσι, χρησιμοποιώντας της συναρτήσεις για να αναγνωρίσει το δοσμένο input. Συνεχίζοντας, το πρόγραμμα μέσω της *factor_tail()* αν εντοπίσει το not τότε συνεχίζει την ανάγνωση. Αν όχι, τότε προχωράει μέσω της *expr()* στο *term_tail(), και* εκεί βλέπει αν υπάρχει κάποιο από τα or, and. Καταλυτικό ρόλο στο πρόγραμμα παίζουν οι συναρτήσεις *match(), next_token(),* οι οποίες μέσω του προγράμματος, όταν αυτό διαβάζει έναν σωστό δοσμένο χαρακτήρα, τον κάνει match με κάποιο από τα σύμβολα που έχουμε βάλει από την αρχή στο λεξικό του προγράμματος, και έπειτα φέρνει τον επόμενο προς ανάγνωση χαρακτήρα, μέσω της *next_token*, η οποία "σκανάρει" και τον διαβάζει.

- **<u>Ενδεικτικά Στιγμιότυπα Δοκιμών:</u>**


Τερματικό
Αρχείο Επεξεργασία Προβολή Αναζήτηση Τερματικό Βοήθεια
self.term()
File "parser.py", line 97, in term
self.factor()
File "parser.py", line 115, in factor
self.expr()
File "parser.py", line 81, in expr
self.term_tail()
File "parser.py", line 87, in term_tail
self.aoop()
File "parser.py", line 130, in aoop
self.match('and')
File "parser.py", line 51, in match
self.la, self.val = self.next_token()
File "parser.py", line 56, in next_token
return self.scanner.read()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 94, in read
self.text, action = self.scan_a_token()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 138, in scan_a_token
raise errors.UnrecognizedInput(self, self.state_name)
plex.errors.UnrecognizedInput: '', line 2, char 16: Token not recognised in state ''
aristotle@reggaeshark ~/Desktop/ex2 $
data.txt (~/Desktop/ex2)
Αρχείο Επεξεργασία Προβολή Αναζήτηση Εργαλεία Έγγραφα Βοήθεια
data.txt
a = true or false
b = 0 or (1 and 4)
c = T and (T and not F)
Απλό κείμενο
Πλάτος Tab: 4
Γρ 2, Στηλ. 18
ΕΙΣ
Μενού
deadm...
~/Desk...
| Τερμ...
ex2
| data.t...
Δίκτυο
gram...
02:32


data.txt (~/Desktop/ex2)
Αρχείο Επεξεργασία Προβολή Αναζήτηση Εργαλεία Έγγραφα Βοήθεια
parser.py
data.txt
a = true or false
b = 0 or (1 and 0)
c = T and (T and not F)
d = a or b
Απλό κείμενο
Πλάτος Tab: 4
Γρ 4, Στηλ. 13
ΕΙΣ
Τερματικό
Αρχείο Επεξεργασία Προβολή Αναζήτηση Τερματικό Βοήθεια
self.factor()
File "parser.py", line 115, in factor
self.expr()
File "parser.py", line 81, in expr
self.term_tail()
File "parser.py", line 87, in term_tail
self.aoop()
File "parser.py", line 130, in aoop
self.match('and')
File "parser.py", line 51, in match
self.la, self.val = self.next_token()
File "parser.py", line 56, in next_token
return self.scanner.read()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 94, in read
self.text, action = self.scan_a_token()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 138, in scan_a_token
raise errors.UnrecognizedInput(self, self.state_name)
plex.errors.UnrecognizedInput: '', line 2, char 16: Token not recognised in state ''
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
aristotle@reggaeshark ~/Desktop/ex2 $
Μενού
[Recursive descent...
Documents
| data.txt (~/Deskt...
| Τερματικό
03:36


Τερματικό
Αρχείο Επεξεργασία Προβολή Αναζήτηση Τερματικό Βοήθεια
File "parser.py", line 146, in <module>
parser.parse(fp)
File "parser.py", line 47, in parse
self.stmt_list()
File "parser.py", line 60, in stmt_list
self.stmt()
File "parser.py", line 70, in stmt
self.match('=')
File "parser.py", line 51, in match
self.la, self.val = self.next_token()
File "parser.py", line 56, in next_token
return self.scanner.read()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 94, in read
self.text, action = self.scan_a_token()
File "/usr/local/lib/python3.4/dist-packages/plex/scanners.py", line 138, in scan_a_token
raise errors.UnrecognizedInput(self, self.state_name)
plex.errors.UnrecognizedInput: '', line 1, char 4: Token not recognised in state ''
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
Match Error
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
aristotle@reggaeshark ~/Desktop/ex2 $
data.txt (~/Desktop/ex2)
Αρχείο Επεξεργασία Προβολή Αναζήτηση Εργαλεία Έγγραφα Βοήθεια
data.txt
a = true or false
b = 0 or (1 and 1)
c = T and (T not F)
Απλό κείμενο
Πλάτος Tab: 4
Μενού
02:32


Τερματικό
Αρχείο Επεξεργασία Προβολή Αναζήτηση Τερματικό Βοήθεια
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
Match Error
aristotle@reggaeshark ~/Desktop/ex2 $ python3 parser.py
Match Error
aristotle@reggaeshark ~/Desktop/ex2 $
data.txt (~/Desktop/ex2)
Αρχείο Επεξεργασία Προβολή Αναζήτηση Εργαλεία Έγγραφα Βοήθεια
data.txt
p
Απλό κείμενο
Πλάτος Tab: 4
Γρ 2, Στηλ. 1
ΕΙΣ
Μενού
02:31