

ΜΕΤΑΓΛΩΤΤΙΣΤΕΣ 2019 (ΕΡΓΑΣΙΑ 2)

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: Ζαμπούνης Χρήστος

ΑΜ: Π2015079

- Κανόνες γραμματικής:

Grammar	
Stmt_list	→ Stmt Stmt_list .
Stmt	→ id equal Expr print Expr. .
Expr	→ Term Term_tail.
Term_tail	→ xor Term Term_tail .
Term	→ Factor Factor_tail.
Factor_tail	→ or Factor Factor_tail .
Factor	→ Atom Atom_tail.
Atom_tail	→ and Atom Atom_tail .
Atom	→ lparen Expr rparen id number.

- **Αποτέλεσμα 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 Expr Stmt_list Stmt.
- No cycles.

nonterminal	first set	follow set	nullable	endable
Stmt_list	id print	\emptyset	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
Expr	lparen id number	rparen id print	no	yes

The grammar is LL(1).

Άρα οι κανόνες διαβάζονται από αριστερά προς δεξιά και διατίθεται μονοπάτι προς όλα τα μη τερματικά σύμβολα. Επίσης δεν υπάρχουν κυκλικές μεταβάσεις, οπότε συμφωνεί με την μέθοδο της αναδρομικής κατάβασης

- **Πίνακας με FIRST και FOLLOW set της γραμματικής:**

Μη τερματικά	FIRST	FOLLOW
Stmt_list	id print	ϵ
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

Expr	(id number) id print
------	-------------	------------

- **Αποτελέσματα εξόδου parser.py:**

- **Σωστές τιμές:**

```

25 a = 101 xor 111
26 b = 1001100 and 0001001
27 c = 1001
28 print a
29 print b
30 print c
31 print 1 or 0 xor 1

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 11: bash

```

zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$ python parser.py
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$

```

- **Λανθασμένες τιμές:**

```

14
15 test_variable = 1010101 ^ 1912 and 101 xor 111 and 1100 and
16
17 print(test_variable)

```

PROBLEMS OUTPUT TERMINAL ... 5: bash

```

zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$ python par
ser.py
Traceback (most recent call last):
  File "parser.py", line 153, in <module>
    parser.parse(fp)
  File "parser.py", line 60, in parse
    self.stmt list()
  File "parser.py", line 65, in stmt list
    self.stmt list()
  File "parser.py", line 64, in stmt list
    self.stmt()
  File "parser.py", line 74, in stmt
    self.match("=")
  File "parser.py", line 53, in match
    raise ParseError("found {} instead of {}".format(self.la,token))
main .ParseError: found BIT TOKEN instead of =
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$

```

- Αποτελέσματα εξόδου runner.py:

- Σωστές τιμές:

```
25 a = 101 xor 111
26 b = 1001100 and 0001001
27 c = 1001
28 print a
29 print b
30 print c
31 print 1 or 0 xor 1
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 12: bash

```
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$ python runner.py
10
1000
1001
0
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$
```

- Λανθασμένες τιμές:

```
27 a = 010 | 011
28 b = 1010101 ^ 1912
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL 15: bash

```
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$ python runner.py
Traceback (most recent call last):
  File "runner.py", line 136, in <module>
    parser.parse(fp)
  File "runner.py", line 53, in parse
    self.stmt list()
  File "runner.py", line 57, in stmt list
    self.stmt()
  File "runner.py", line 69, in stmt
    self.st[varname] = self.expr()
  File "runner.py", line 78, in expr
    t = self.term()
  File "runner.py", line 91, in term
    f = self.factor()
  File "runner.py", line 104, in factor
    a = self.atom()
  File "runner.py", line 129, in atom
    self.match('BIT TOKEN')
  File "runner.py", line 47, in match
    self.la,self.text = self.next token()
  File "runner.py", line 43, in next token
    return self.scanner.read()
  File "/home/zabzuki/.local/lib/python3.7/site-packages/plex/scanners.py", line 94, in read
    self.text, action = self.scan a token()
  File "/home/zabzuki/.local/lib/python3.7/site-packages/plex/scanners.py", line 138, in scan a tok
en
    raise errors.UnrecognizedInput(self, self.state name)
plex.errors.UnrecognizedInput: '', line 27, char 8: Token not recognised in state ''
zabzuki@zabzuki-Z97-D3H:~/Desktop/GitHub/python/compilers1819a2$
```

- **Πηγές:**

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

<http://smlweb.cpsc.ucalgary.ca/start.html>