

Ιόνιο Πανεπιστήμιο  
Τμήμα Πληροφορικής



Προγραμματιστική Εργασία #2 στο μάθημα  
Μεταγλωττιστές

Επιβλέπων καθηγητής: Μ. Στεφανιδάκης

Τσίγκα Αναστασία, Π2015049

Κέρκυρα, Ιούνιος 2019

## A) Ζητούμενη γραμματική

```
Stmt_list  -> Stmt Stmt_list | .
Stmt       -> id assign 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       -> leftPar Expr rightPar | id | number.
```

## B) Έλεγχος συμβατότητας LL(1) & FIRST, FOLLOW sets

**Grammar**

```
Stmt_list → Stmt Stmt_list
           | .
Stmt       → id assign 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       → leftPar Expr rightPar
           | id
           | number.
```

Some sentences generated by this grammar: {ε, print id, print number, id assign id, print id and id, id assign number, print id and number, print number and id, id assign id and id, id assign number and id, print number and number, id assign id and number, id assign id and id and id, id assign number and number, id assign id and id and number, id assign id and number and id, id assign number and id and id, id assign id and number and number, id assign number and id and number, id assign number and number and id}

- 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	∅	yes	yes
Stmt	id print	id print	no	yes
Term_tail	xor	rightPar id print	yes	yes
Term	leftPar id number	rightPar xor id print	no	yes
Factor_tail	or	rightPar xor id print	yes	yes
Factor	leftPar id number	rightPar or xor id print	no	yes
Atom_tail	and	rightPar or xor id print	yes	yes
Atom	leftPar id number	rightPar and or xor id print	no	yes
Expr	leftPar id number	rightPar id print	no	yes

The grammar is LL(1).

## Γ) Πηγές

1. Παράδειγμα: <https://gist.github.com/mixstef/946fce67f49f147991719bfa4d0101fa>
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, *Μεταγλωττιστές: Αρχές, Τεχνικές & Εργαλεία*, Επιμέλεια ελληνικής έκδοσης Π. Αλεφραγκής, Α. Βώρος, Ν. Βώρος, Κ. Μασσέλος, Εκδόσεις Νέων Τεχνολογιών, Έκδοση Δεύτερη, 2014
3. Context Free Grammar Tool: <http://smlweb.cpsc.ucalgary.ca/start.html>