# **Σύγκριση Concrete Syntax Tree και Abstract Syntax Tree**

Θα παρουσιάσουμε μια σύγκριση μεταξύ του CST και του AST που παρήχθησαν κατά τη δεύτερη φάση και την τρίτη φάση της εργασίας, δηλαδή πριν και μετά την προσθήκη του τμήματος AST στο αρχείο grammar.

Για την σύγκριση, όπως και κατά το testing χρησιμοποιούμε το παρακάτω εκτενές αρχείο python:

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Τρέχοντας τα αρχεία Parser2Test και ASTTest1 (ή ASTTest1 και για τα δύο examples) παίρνουμε το παρακάτω output:

**Concrete Syntax Tree**

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Στη συνέχεια το AST που κατασκευάζεται είναι το εξής:

**Abstract Syntax Tree**

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

Από τις παραπάνω εικόνες φαίνεται πόσο πολύ έχει συμπυκνωθεί το δέντρο που κατασκευάζεται. Καταρχάς δεν εμφανίζονται περιττά tokens, όπως παρενθέσεις, τελεστές κ.α. Παράλληλα βλέπουμε ότι πολλές εντολές έχουν αναχθεί σε πιο γενικές (όπως τα statements, something, multiplication, singleArrayExpression κτλ).

Γενικά η δομή του τμήματος AST στη γραμματική προσομοιώνει όσο το δυνατόν περισσότερο τη δομή του BNF αρχείου, επομένως σχεδόν όλες οι κανόνες και οι εναλλακτικές τους έχουν αναχθεί στον γενικότερο κανόνα που μπορεί να ενθυλακώσει όλες τις εναλλακτικές μαζί. Μερικοί κανόνες όπως οι Identifier και CommaIdAsId έχουν παραμείνει. είτε γιατί παρατηρήσαμε ότι άλλοι κανόνες μπορούν να αναχθούν σε αυτούς (π.χ.. as\_id σε identifier) είτε οι ίδιοι χρησιμεύουν σαν βοηθητικοί και δεν μπορούν να αναχθούν σε γενικότερους χωρίς να προστεθούν επιπλέον κανόνες (όπως συμβαίνει με το comma\_id\_as\_id)