

# COMPILERS 2021

by: Βάιος Λύτρας

## Main

Στην συνάρτηση main υπάρχει μόνο το κάλεσμα των visitor και η εκτύπωση των offset. Υπάρχει επίσης σχολιασμένο κομμάτι κώδικα που εκτυπώνει όλα τα περιεχόμενα του symbol table που το χρησιμοποιούσα και αν θέλετε μπορείτε να το χρησιμοποιήσετε για να δείτε παραπάνω πράγματα. Έχω βάλει και μία global μεταβλητή error η οποία γίνεται true αν προκύψει κάποιο error σε κάποιον από τους visitor έτσι ώστε να εκτυπωθούν τα κατάλληλα μηνύματα.

## Symbol Table Visitor

Για το symbol table έχω χρησιμοποιήσει 4 δομές από τις οποίες η μία είναι βασική και οι 3 για επιπλέον πληροφορία. Η βασική δομή είναι:

```
LinkedHashMap<String, LinkedHashMap<String, List<String>>> ClassMap
```

Το εξωτερικό map κρατάει τις κλάσεις του προγράμματος σε string το οποίο συνδέεται με ένα map το οποίο κρατάει τις συναρτήσεις αυτής της κλάσης και κάθε συνάρτηση έχει μια λίστα με τις μεταβλητές της αλλά και μαζί με τον τύπο τους πχ. int[] var. Εδώ να σημειωθεί ότι σε αυτή τη δομή δεν αποθηκεύονται οι τύποι επιστροφής των συναρτήσεων. Η δεύτερη δομή είναι η:

```
LinkedHashMap<String, List<String>> MethodArgs
```

Η οποία κρατάει τα ορίσματα της κάθε συνάρτησης αλλά για να ξεχωρίζονται συναρτήσεις με το ίδιο όνομα έχουν μπροστά το όνομα της κλάσης που ανήκουν και τον τύπο επιστροφής τους.

Πχ: int MyClass.MyFunction. Η Τρίτη δομή είναι η:

```
LinkedHashMap<String, List<String>> ClassVars;
```

Μέσα στην οποία κρατούνται οι μεταβλητές των κλάσεων μαζί με τον τύπο τους. Η τελευταία δομή:

```
LinkedHashMap<String, String> Inheritance;
```

Η οποία κρατάει την κληρονομικότητα στο στυλ <γονέας, παιδί>.

Μέσα στο symbol table visitor έχω κάνει override ότι έχει declaration και κάποια έξτρα που χρειάζονται έτσι ώστε να στηθεί όλο το symbol table και επίσης εκεί μέσα γίνονται όλοι οι έλεγχοι για κλάσεις με το ίδιο όνομα ή μεταβλητές ή συναρτήσεις με το ίδιο όνομα ή να κάνεις extend σε κλάση που δεν υπάρχει κλπ. Το πως τσεκάρονται αυτά υπάρχουν τα αντίστοιχα μηνύματα στο output που γενικά είναι αρκετά διευκρινηστικά οπότε μπορεί κάποιος να καταλάβει τι έλεγχο κάνει το αντίστοιχο κομμάτι κώδικα βάσει αυτού του μηνύματος. Το ίδιο ισχύει και για τον type check visitor με τα μηνύματα των error.

# Type Check Visitor

---

Το `TypeCheckVisitor` κάνει `extend` το `SymbolTableVisitor` και αντιγράφει όλα τα δεδομένα του `symbol table`. Επίσης όλες οι συναρτήσεις `visit` αυτού του visitor παίρνουν σαν όρισμα ένα `String argu` το οποίο είναι η κλάση και η μέθοδος μέσα στην οποία βρισκόμαστε ή αλλιώς το `scope` (πχ: `"MyClass foo"`) το οποίο χρησιμεύει στο `type checking`. Ξεκινώντας από το χαμηλότερο επίπεδο τα `PrimaryExpression` επιστρέφουν μηνύματα όπως `int`, `Boolean`, `this`, `allocation`, `arrayallocation` και `identifier`. Αυτά χρησιμοποιούνται από πιο το πιο πάνω επίπεδο `Expression` έτσι ώστε να κάνει σωστά τους ελέγχους. Πχ:

`CompareExpression ::= PrimaryExpression "<" PrimaryExpression`

Αν τα δύο `PrimaryExpression` επιστρέψουν `int` τότε δεν υπάρχει `error`. Αν επιστρέψουν οτιδήποτε από τα υπόλοιπα υπάρχει `error` εκτός και αν είναι `identifier`. Αν πάρουμε `this` τότε μετατρέπουμε το `this` στο όνομα της κλάσης στην οποία βρισκόμαστε μέσω του ορίσματος `argu`. Αν είναι `identifier` έρχεται στο παιχνίδι μία πολύ χρήσιμη `utility function` που έφτιαξα και λέγεται `var_exists`. Η `var_exists` παίρνει σαν όρισμα έναν `identifier` και την μέθοδο μέσα στην οποία θέλουμε να την χρησιμοποιήσουμε και την κλάση στην οποία ανήκει αυτή η μέθοδος. Η συνάρτηση αυτή ψάχνει στο `symbol table` αν η μεταβλητή ανήκει στην μέθοδο ή στα `arguments` της μεθόδου ή στις μεταβλητές της κλάσης. Αν κάπου την βρει τότε επιστρέφει τον τύπο της αλλιώς βγάζει `error`. Οπότε αν το `PrimaryExpression` επιστρέψει `identifier` καλούμε αυτή την συνάρτηση και μετατρέπουμε το `identifier` σε τύπο. Όλοι οι έλεγχοι γίνονται κατά αυτόν τον τρόπο επίσης υπάρχουν και αντίστοιχοι έλεγχοι για `custom` τύπους. Αυτή είναι η γενική ιδέα πίσω από την υλοποίηση του `type check visitor` μου.

# Compilation-Run

---

`make compile`

`java Main.java [file1] [file2] . . . [fileN]`

Καλό είναι τα παραδείγματα να είναι μέσα στον ίδιο φάκελο με την `Main` (`minijava_example`).