

ΜΕΤΑΓΛΩΤΤΙΣΤΕΣ

ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2020

3Η ΕΡΓΑΣΙΑ

BIPBIDAKH ΧΡΥΣΟΥΛΑ

AM: 1115201600021

Εντολές μεταγλώττισης:

```
make
```

Εντολές εκτέλεσης για ένα αρχείο:

```
java Main file1.java
```

```
/home/users/thp06/clang/clang -o out file1.ll -Wno-override-module
```

```
./out
```

Στην εργασία αυτή θεωρούμε ως δεδομένο ότι τα αρχεία που παίρνουμε ως είσοδο θα περνούσαν με επιτυχία από τον `TypeCheckVisitor` της δεύτερης εργασίας, δηλαδή ότι δεν θα έχουν ούτε συντακτικά ούτε σημασιολογικά λάθη. Για κάθε ένα αρχείο τρέχουμε αρχικά τον `DefCollectVisitor` της δεύτερης εργασίας ούτως ώστε να δημιουργηθούν οι απαραίτητες δομές `SymbolTable`, `offsets`. Έπειτα καλούμε τον `Third_Part_Visitor`, ο οποίος σκανάρει ξανά όλο το αρχείο και παράλληλα μετατρέπει τις εντολές σε γλώσσα LLVM, την οποία γράφει στο αρχείο `file1.ll` (έστω `file1.java` το αρχείο εισόδου).

`value_flag`: Την μεταβλητή αυτή τη χρησιμοποιώ για να μπορώ να ξέρω όταν για παράδειγμα θέλω να κάνω `ArrayLookup`, αν σκοπός μου είναι να διαβάσω την τιμή αυτού το πεδίου (`value_flag = "right"`), ή αν θέλω να αποκτήσω πρόσβαση στη διεύθυνση αυτής της μεταβλητής για να αποθηκεύσω για παράδειγμα μια τιμή σε αυτήν (`value_flag = "left"`).

`Var_of_Function`: Η συνάρτηση αυτή είναι αρκετά σημαντική γιατί με βοηθάει να βρω για μία μεταβλητή εάν είναι δηλωμένη μέσα στο σώμα μιας συνάρτησης, εάν είναι όρισμα της συνάρτησης, αν είναι πεδίο της κλάσης στην οποία ανήκει η συνάρτηση, ή αν τέλος είναι πεδίο κάποιας υπερκλάσης της κλάσης στην οποία ανήκει η συνάρτηση και σε ποιά. Επιθυμώ να ξέρω αυτήν την πληροφορία για να ξέρω ποιος θα είναι ο παραγόμενος κώδικας σε LLVM, αν αρκεί δηλαδή ένα απλό `load` για να φορτώσω την τιμή της

μεταβλητής αυτής, ή αν θα πρέπει να κάνω `getelementptr` προκειμένου να φτάσω στην μεταβλητή.

VTable: Σε αυτή τη δομή αποθηκεύω το VTable της κάθε κλάσης. Για κάθε νέα κλάση που επιθυμώ να φτιάξω το VTable της, αρχικά ελέγχω αν έχει κάποια υπερκλάση. Αν έχει τότε βρίσκω το VTable της και το αντιγράφω σε αυτό της νέας κλάσης, προσέχοντας όμως αν η υποκλάση έχει κάνει `override` κάποια συνάρτηση, τότε να δηλωθεί στο VTable ως συνάρτηση της υποκλάσης. Αφού μεταφέρω όλη την πληροφορία προσθέτω στη συνέχεια τις υπόλοιπες συναρτήσεις της υποκλάσης εφόσον έχει.

Boolean: Για την διαχείριση των boolean μεταβλητών έχω χρησιμοποιήσει τις συναρτήσεις `zext` και `trunc` ούτως ώστε να μπορώ να μετατρέπω τους τύπους από `i1` σε `i8` και το αντίστροφο.