

Γιώργος Παπαδόπουλος (Αρ. Μητρώου: 3160131)

Δικαία Σωτηροπούλου (Αρ. Μητρώου: 3160172)

Αλέξανδρος Ψαραύτης (Αρ. Μητρώου: 3160196)

Β' Φάση Project

1. Χρήση μη δηλωμένης μεταβλητής

Στον *myvisitor* δημιουργούμε το *Hashtable variables* και εκχωρούμε σε αυτόν όλες τις μεταβλητές που εμφανίζονται στο αρχείο της *minipython*. Έπειτα στον *myvisitor2* δημιουργούμε τη μέθοδο *checkVariableDefinition* η οποία ελέγχει αν το *node* που έχει δοθεί ως όρισμα υπάρχει μέσα στο *Hashtable operations*. Ο *Hashtable operations* έχει τις ίδιες καταχωρήσεις με τον *Hashtable variables* αλλά επιπλέον κρατάει και τύπους πράξεων (Περισσότερα παρακάτω). Αν το *node* αυτό δεν υπάρχει στο *operations* τότε καλείτε η συνάρτηση *notDefined* και εμφανίζεται μήνυμα.

2. Κλήση μη δηλωμένης συνάρτησης

Στον *myvisitor* κάνουμε *Override* τη μέθοδο *inAFunction* ώστε όταν ορίζεται μια καινούργια συνάρτηση στη *minipython*, την κρατάμε στην *curFunc*. Η *curFunc* είναι τύπου *FunctionData* και η κλάση αυτή έχει δημιουργηθεί για να κρατάει τα στοιχεία μιας συνάρτησης (το όνομα της, τα ορίσματα της και τον τύπο επιστροφή της). Μπαίνοντας στην *inAFunction* αποθηκεύουμε το όνομα της συνάρτησης στην *curFunc*. Στη συνέχεια κάνουμε *override* τις μεθόδους *inASingleArgument* (παράμετρος με *default* τιμή) και *inASingleArgument* (απλή παράμετρος) και αντίστοιχα αποθηκεύουμε τα ονόματα των παραμέτρων και τους τύπους τους στην *curFunc*. Τέλος κάνουμε *override* την *outAFunction* (έξοδος από την δήλωση συνάρτησης) οπότε και αποθηκεύουμε την *curFunc* στο *Hashtable functions* το οποίο κρατάει όλες τις συναρτήσεις του κώδικα με όλες τις απαραίτητες πληροφορίες για αυτές.

Στον *myvisitor2* κάνουμε *Override* τη συνάρτηση *outAFunctionCall*, η οποία θα καλέσει την *getFunctionData* όπου θα ελεγχθεί αν υπάρχει το όνομα της συνάρτησης που έχει κληθεί ή όχι. Αν δεν υπάρχει, καλείται η *notDefined* και τυπώνεται κατάλληλο μήνυμα λάθους.

3. Λάθος ορισμός ορισμάτων σε κλήση συνάρτησης

Στον *myvisitor2* ακολουθούμε αντίστοιχη στρατηγική με αυτή για τη δημιουργία αντικειμένων *FunctionData* (που αναφέρθηκε στο 2), για τη δημιουργία αντικειμένων *functionCalls*, τα οποία κρατάνε πληροφορίες για την συνάρτηση που καλείται και τα ορίσματα που περνιούνται. Στη συνάρτηση *getFunctionData*, αν η συνάρτηση της *minipython* έχει οριστεί (άρα είναι μέσα στο *Hashtable functions*), τότε ψάχνουμε αν υπάρχει άλλη συνάρτηση με το ίδιο όνομα αλλά με διαφορετικό πλήθος ορισμάτων (αν έχει γίνει *overload* της συνάρτησης). Αν δε βρεθεί τέτοια συνάρτηση, τότε τυπώνεται κατάλληλο μήνυμα λάθους.

4. Χρήση μεταβλητής διαφορετικού τύπου ως ακέραιο

Στον *myvisitor2* δημιουργούμε τη μέθοδο *getOperationType* η οποία ελέγχει τον τύπο των μεταβλητών που υπάρχουν μεταξύ των συμβόλων για αριθμητικές πράξεις, (π.χ. αν έχουμε την εντολή $a + b$, ελέγχει τι τύπου μεταβλητή είναι η a και τι τύπου είναι η b). Αν οι μεταβλητές a και b είναι *Numeric* τότε μπορεί να γίνει η πράξη κανονικά. Αν είναι τύπου *NAN*, τότε δεν γνωρίζουμε ακριβώς τι τύπου είναι. Αν οι a και b είναι διαφορετικού τύπου

Γιώργος Παπαδόπουλος (Αρ. Μητρώου: 3160131)
Δικαία Σωτηροπούλου (Αρ. Μητρώου: 3160172)
Αλέξανδρος Ψαράυτης (Αρ. Μητρώου: 3160196)

μεταξύ τους ή είναι διαφορετικοί των παραπάνω, τότε η πράξη δε μπορεί να γίνει και εμφανίζεται κατάλληλο μήνυμα λάθους.

Η *checkSingleOperations* ελέγχει αν η μεταβλητή στην οποία πάει να γίνει η πράξη *++* ή *--* είναι τύπου *Numeric*. Αν δεν είναι, τότε σημαίνει ότι η πράξη αυτή δε μπορεί να γίνει, άρα θα τυπωθεί κατάλληλο μήνυμα λάθους.

5. Πράξεις μεταξύ μεταβλητών *none*

Ο έλεγχος αυτός πραγματοποιείται στον παραπάνω έλεγχο (4), δηλαδή στον *myvisitor2* με την κλήση της συνάρτησης *getOperationType*. Αν οι μεταβλητές με τις οποίες θέλουμε να κάνουμε κάποια αριθμητική πράξη είναι τύπου *none* τότε εμφανίζεται κατάλληλο μήνυμα λάθους.

6. Χρήση *expression* σε λογική πράξη

Ο συγκεκριμένος έλεγχος πραγματοποιείται μέσω του *grammar* και του πώς έχει οριστεί το *production statement*.

7. Λάθος τρόπος χρήσης συνάρτησης

Ο έλεγχος αυτός πραγματοποιείται στον *myvisitor2* με την κλήση της συνάρτησης *getOperationType* μέσω της οποίας θα βρούμε τον τύπο επιστροφής που έχει μια συνάρτηση. Αν η συνάρτηση της *miniptyhon* επιστρέφει κάτι που δεν είναι *numeric* (άρα δε μπορούν να γίνουν πράξεις) τότε τυπώνεται μήνυμα λάθους.

8. Επανάληψη δήλωσης συνάρτησης με τον ίδιο αριθμό ορισμάτων

Στον *myvisitor* ορίζουμε τη μέθοδο *alreadyDefined* η οποία καλείται από την *outAfunction*. Αν βρούμε ότι η συνάρτηση που ελέγχουμε τώρα έχει καταχωρηθεί στο *Hashtable functions* (άρα έχει οριστεί ξανά), τότε ελέγχουμε αν έχουν διαφορετικό πλήθος ορισμάτων. Αν ο αριθμός των ορισμάτων είναι ίδιος τότε τυπώνεται κατάλληλο μήνυμα λάθους.