

Μεταγλωττιστές 2023

Προγραμματιστική Εργασία #3

Ζητούμενο

Ο στόχος της άσκησης είναι να τροποποιήσετε τη **γραμματική των αριθμητικών εκφράσεων και εντολών ανάθεσης/εκτύπωσης** που έχετε χρησιμοποιήσει στο εργαστήριο, έτσι ώστε να προσθέσετε υποστήριξη για τους αριθμητικούς τελεστές **%** (modulo) και **^** (ύψωση σε δύναμη). Στη συνέχεια θα πρέπει να κατασκευάσετε **συντακτικό αναλυτή** αναδρομικής κατάβασης (recursive descent parser) για τη νέα γραμματική, κώδικα για τη δημιουργία του αντίστοιχου **AST (abstract syntax tree)** και πρόγραμμα εκτέλεσης (interpreter) για το AST που προκύπτει.

Οι προτεραιότητες και προσεταιριστικότητα (associativity) των νέων τελεστών θα πρέπει να είναι ως εξής:

Προτεραιότητα τελεστών (από την υψηλότερη στη χαμηλότερη)

\wedge
$*, /, \%$
$+, -$

Προσεταιριστικότητα τελεστών

\wedge	δεξιιά
$*, /, \%$	αριστερή
$+, -$	αριστερή

Βήματα υλοποίησης

1. Θα βρείτε τη γραμματική των αριθμητικών εκφράσεων και εντολών ανάθεσης/εκτύπωσης στο: <https://mixstef.github.io/courses/compilers/parsing-notes2.pdf> (σελ. 29).

Προσθέστε τους κανόνες για την υποστήριξη των τελεστών **%** και **^**. Θυμηθείτε ότι η προτεραιότητα κάθε τελεστή καθορίζει σε ποιο επίπεδο της γραμματικής θα βρίσκεται (όσο μεγαλύτερη η προτεραιότητα, τόσο «προς τα κάτω» βρίσκεται ο αντίστοιχος κανόνας). **Διατηρήστε την ιδιότητα LL(1)** για τη νέα γραμματική (μπορείτε να χρησιμοποιήσετε κάποιο εργαλείο διαθέσιμο on-line για τον έλεγχο όπως π.χ. το <http://smlweb.cpsc.ucalgary.ca/start.html>).

2. Βρείτε τα **FIRST** και **FOLLOW sets** για τη νέα γραμματική και τροποποιήστε **τον συντακτικό αναλυτή αναδρομικής κατάβασης** που έχετε δουλέψει στο εργαστήριο. Αν δεν τον έχετε διαθέσιμο, μπορείτε να χρησιμοποιήσετε ως βάση το:

<https://mixstef.github.io/courses/compilers/arithmetic-parser.ipynb>

3. Προσθέστε κώδικα για την **κατασκευή του δέντρου AST** για να καλύψετε τους νέους τελεστές. Δείτε και τις οδηγίες στο <https://mixstef.github.io/courses/compilers/ast-building.pdf>. Εάν ξεκινάτε τη

συγγραφή κώδικα από την αρχή, χρησιμοποιήστε το υπόδειγμα από το:

<https://mixstef.github.io/courses/compilers/parser-ast-builder-interpreter-template.ipynb>

4. Ελέγξτε εάν το παραγόμενο AST εκφράζει σωστά την προσεταιριστικότητα των τελεστών. Εάν όχι, θα πρέπει να τροποποιήσετε τον κώδικα του συντακτικού αναλυτή, σύμφωνα με το:

<https://mixstef.github.io/courses/compilers/rd-tweaks.pdf>

5. Προσθέστε τις μεθόδους διάσχισης του AST και επίσκεψης των κόμβων του για την εκτέλεση του προγράμματος εισόδου, με την υποστήριξη των νέων τελεστών % και ^.

Παράδοση εργασίας

Ανεβάστε το τελικό σας notebook όπου θα φαίνεται και η έξοδος της εκτέλεσης για συγκεκριμένη είσοδο (ως αρχείο zip) στο opencourses (**Εργασία 3**) έως και τη **Δευτέρα 5/6**.