



Γλώσσες Προγραμματισμού II

Οι ασκήσεις πρέπει να παραδοθούν στους διδάσκοντες σε ηλεκτρονική μορφή μέσω του συνεργατικού συστήματος ηλεκτρονικής μάθησης moodle.softlab.ntua.gr. Η προθεσμία παράδοσης θα τηρείται αυστηρά. Έχετε δικαίωμα να καθυστερήσετε το πολύ μία άσκηση.

Άσκηση 3 Συστήματα τύπων — σημασιολογία οκνηρής αποτίμησης

Προθεσμία παράδοσης: 9/1/2013

Η Standard ML και οι διάφορες παραλλαγές της (OCaml, F#) έχουν *πρόθυμη σημασιολογία* (eager, strict, call-by-value), εν αντιθέσει π.χ. με τη Haskell που έχει *οκνηρή σημασιολογία* (lazy, non-strict, call-by-need). Στις πρώτες όμως γλώσσες, είναι δυνατόν να υλοποιηθεί οκνηρή σημασιολογία, ενίοτε και με την υποστήριξη μίας κατάλληλης βιβλιοθήκης. Ας προσπαθήσουμε να επεκτείνουμε τη γλώσσα που περιγράφεται στις διαφάνειες των παραδόσεων της 31/10 και της 28/11 προς αυτή την κατεύθυνση.

Έστω η γλώσσα που περιέχει: *απλούς τύπους* (διαφάνειες 4–12), *συναρτήσεις* με σημασιολογία call-by-value (διαφάνειες 13–15), όποιες και όσες από τις *επεκτάσεις* που συζητήσαμε θέλετε, αλλά πάλι με σημασιολογία call-by-value (διαφάνειες 16–31), και *αναφορές* (διαφάνειες 32–39).

Προσθέτουμε στη σύνταξη της γλώσσας δύο νέους τελεστές:

$$e ::= \dots \mid \mathbf{lazy} \ e \mid \mathbf{force} \ e$$

Η επιθυμητή συμπεριφορά των παραπάνω εξηγείται στο παράδειγμα που ακολουθεί. Οι οκνηρές εκφράσεις στα x και y υπολογίζονται μία φορά, ενώ εκείνη στο w καμία. Για τον τελευταίο πολλαπλασιασμό, είναι σημαντικό ότι η αποτίμηση γίνεται από αριστερά προς τα δεξιά.

```
let r = ref 1 in
let x = lazy (r := !r + 1; !r) in
let f = λz. force z * (force z + 1) in
let y = lazy (f x) in
let w = lazy (r := 1234) in
!r * f y
```

— αποτιμάται σε 42

Ζητούνται:

- (Προθέρμανση:) Η υλοποίηση του Κόσκινου του Ερατοσθένη ως άπειρη λίστα σε μία εκ των Standard ML, OCaml ή F#, χρησιμοποιώντας τους τελεστές **lazy** και **force** ή τους αντίστοιχούς τους. Μία υλοποίηση σε Haskell είναι διαθέσιμη στη σελίδα του μαθήματος (παράδοση 10/10).
- Οι κανόνες τύπων για τους νέους τελεστές.
- Οι κανόνες που ορίζουν τη λειτουργική σημασιολογία των νέων τελεστών ή (εναλλακτικά) ο ορισμός τους ως παραγομένων μορφών (derived forms, syntactic sugar).

Δώστε ιδιαίτερη προσοχή στην ακρίβεια των τυπικών ορισμών σας (2 και 3).

Αν χρησιμοποιήσετε σχετικές εργασίες τρίτων που βρήκατε π.χ. στο διαδίκτυο, μην ξεχάσετε να αναφέρετε λεπτομερώς τις πηγές σας.