



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 10 Δηλωτική σημασιολογία

Προθεσμία παράδοσης: 25/3/2008

Έστω η προστακτική γλώσσα προγραμματισμού της Ενότητας 6.4, σελ. 81, των σημειώσεων (ας την ονομάσουμε WHILE). Η δηλωτική σημασιολογία της WHILE ορίζεται στο Κεφάλαιο 6 των σημειώσεων, χρησιμοποιώντας μερικές συναρτήσεις μεταξύ συνόλων. Μία λίγο διαφορετική μορφή της ίδιας σημασιολογίας, βασισμένη στη θεωρία πεδίων, ορίζεται στις διαφάνειες 29–31 της παρουσίασης της 14/2/2008 με τίτλο “Δηλωτική σημασιολογία”.

1. Κατασκευάστε ένα διερμηνέα για τη WHILE, στη γλώσσα προγραμματισμού της αρεσκείας σας, βασισμένο σε μία άμεση υλοποίηση της δηλωτικής σημασιολογίας της WHILE. Ο διερμηνέας σας, αφού εκτελέσει ένα πρόγραμμα WHILE, πρέπει να εκτυπώνει την τελική τιμή της μεταβλητής `result`.

Όσοι επιλέξετε να υλοποιήσετε το διερμηνέα σε Haskell, μπορείτε να ολοκληρώσετε το πρόγραμμα της παρουσίασης της 7/2/2008 (διαθέσιμο από τη σελίδα του μαθήματος). Θα σας είναι χρήσιμος ο τελεστής σταθερού σημείου `fix :: (a -> a) -> a`, που στη Haskell μπορεί απλά να οριστεί ως:

```
fix f = f (fix f)
```

λόγω της οκνηρής αποτίμησής. Π.χ., το παρακάτω δίνει αποτέλεσμα 5040 (= 7!).

```
let factorial = fix (\f -> \n -> if n == 0 then 1 else n * f (n-1))  
in factorial 7
```

2. Μάθετε τι είναι οι *συνέχειες* (continuations) και πώς μπορούν να χρησιμοποιηθούν για τον ορισμό της δηλωτικής σημασιολογίας των γλωσσών προγραμματισμού. Για το σκοπό αυτό, διαβάστε την εργασία του Tennent [5] περί δηλωτικής σημασιολογίας. Οι συνέχειες και η σημασιολογία με συνέχειες καλύπτονται στην §6.

Άλλες βιβλιογραφικές αναφορές που ίσως βρείτε χρήσιμες είναι οι ακόλουθες. Μάλλον δε θα χρειαστεί να τις διαβάσετε, καταφύγετε όμως σε αυτές αν κρίνετε ότι η εργασία του Tennent [5] δε σας επαρκεί ή αν θέλετε να εμβαθύνετε στο θέμα.

- Το (εκτενέστερο) κεφάλαιο του Tennent [7] στο *Handbook of Logic in Computer Science*, που καλύπτει παρόμοια ύλη.
- Η εργασία των Strachey και Wadsworth [4], που είναι η κλασική αναφορά για τη σημασιολογία με συνέχειες.

- Η εργασία των Sethi και Tang [3], που περιγράφει τη σημασιολογία με συνέχειες για μία γλώσσα με πέρασμα τιμών κατ' αξία.
- Γενικότερα και εκτενέστερα κείμενα περί (δηλωτικής) σημασιολογίας, που καλύπτουν και τις συνέχειες, όπως στα βιβλία του Tennent [6] και του Reynolds [2].
- Κείμενα που περιγράφουν τις συνέχειες ως αφαιρέσεις ροής ελέγχου, όπως η εργασία του Thielecke [8] και τα σχετικά κεφάλαια του βιβλίου των Friedman κ.α. [1].

Επίσης, αν τα βρείτε σκούρα, το google είναι φίλος σας... (Δε χρειάζεται να παραδώσετε τίποτα σε αυτό το ερώτημα.)

3. Ορίστε μία δηλωτική σημασιολογία για τη γλώσσα WHILE, βασισμένη σε συνέχειες.
4. Τροποποιήστε το διερωτηθέν που κατασκευάσατε στο ερώτημα (1) ώστε να υλοποιεί άμεσα τη σημασιολογία του ερωτήματος (3).

Βιβλιογραφία

- [1] Daniel P. Friedman, Mitchell Wand, and Christopher T. Haynes. *Essentials of Programming Languages*. MIT Press, 2nd edition, 2001.
- [2] J. C. R. Reynolds. *Theories of Programming Languages*. Cambridge University Press, 1998.
- [3] R. Sethi and A. Tang. Constructing call-by-value continuation semantics. *Journal of the ACM*, 27(3):580–597, July 1980.
- [4] C. Strachey and C. P. Wadsworth. Continuations: A mathematical semantics for handling full jumps. Technical Monograph PRG-11, Oxford University, Programming Research Group, 1974. Reprinted in *Higher Order and Symbolic Computation*, vol. 13, no 1/2, pp. 135–152, 2000, with a foreword by Christopher P. Wadsworth.
- [5] R. D. Tennent. The denotational semantics of programming languages. *Communications of the ACM*, 19(8):437–453, August 1976.
- [6] R. D. Tennent. *Semantics of Programming Languages*. Prentice Hall, 1991.
- [7] R. D. Tennent. Denotational semantics. In *Handbook of Logic in Computer Science*, volume 3: Semantic Structures, pages 169–322. Oxford University Press, 1994.
- [8] Hayo Thielecke. Continuations, functions and jumps. *SIGACT News*, 30(2):33–42, 1999.