



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 Αξιοματική σημασιολογία

Προθεσμία παράδοσης: 1/4/2012

Στην πρόσφατη εξέταση του εργαστηρίου των πρωτοετών δόθηκε το ακόλουθο προγραμματιστικό πρόβλημα (skating):

Ο μικρός Λουκάς παίζει με το πατίνι του. Κουράζεται όμως εύκολα κι έτσι βασίζεται αποκλειστικά στην κλίση του δρόμου. Έστω ότι ο δρόμος όπου κάνει πατίνι ο Λουκάς χωρίζεται σε N μικρά τμήματα και ότι η κλίση καθενός από αυτά παριστάνεται με έναν ακέραιο αριθμό. Αν, για παράδειγμα, ο αριθμός που αντιστοιχεί σε κάποιο τμήμα είναι 3, τότε το τμήμα αυτό είναι κατηφορικό και η ταχύτητα του πατινιού θα αυξηθεί κατά 3 όταν περάσει από εκεί. Αντίθετα, αν ο αριθμός είναι -4 , τότε το τμήμα αυτό είναι ανηφορικό και η ταχύτητα του πατινιού θα μειωθεί κατά 4 όταν περάσει από εκεί. Τα επίπεδα τμήματα παριστάνονται με μηδέν.

Ο Λουκάς μπορεί να διαλέξει από ποιο σημείο του δρόμου θα ξεκινήσει με το πατίνι του. Η αρχική του ταχύτητα είναι μηδενική και θέλει να φτάσει στο τέλος του δρόμου με τη μέγιστη δυνατή ταχύτητα.

Να γράψετε ένα πρόγραμμα το οποίο θα βρίσκει τη μέγιστη δυνατή ταχύτητα στο τέλος του δρόμου.

Παραδείγματα:

- $N = 8$, κλίση: $\{-1, 3, -4, 5, -1, 4, 0, -2\}$, απάντηση: 6.
- $N = 10$, κλίση: $\{5, -6, 7, -8, 14, 12, -11, 9, -5, 4\}$, απάντηση: 23.
- $N = 5$, κλίση: $\{1, 2, 1, 2, -10\}$, απάντηση: 0.

Για να λύσετε το πρόβλημα, υλοποιείτε τη συνάρτηση με την παρακάτω επικεφαλίδα θεωρώντας ότι οι παράμετροι είναι το N και ο πίνακας που περιέχει την κλίση σε κάθε σημείο του δρόμου.

```
int skating (int n, int x[]);
```

Μπορείτε να γράψετε (τουλάχιστον) δύο τέτοιες συναρτήσεις: (α) μία που να υλοποιεί τον προφανή αλγόριθμο, με πολυπλοκότητα $O(N^2)$, και (β) μία που να υλοποιεί κάποιον εξυπνότερο αλγόριθμο, με πολυπλοκότητα $O(N)$. Για τις δύο αυτές συναρτήσεις, αποδείξτε τη μερική ορθότητά τους χρησιμοποιώντας αξιωματική σημασιολογία. Τέλος, σχολιάστε πόσο επηρέασε η ανάγκη κατασκευής της απόδειξης τη μορφή του κώδικα που γράψατε. (Σε έναν ιδανικό κόσμο, η απάντηση θα ήταν: καθόλου!)

Μπορείτε να κάνετε τα παραπάνω με δύο (εναλλακτικούς) τρόπους:

1. Να γράψετε το σώμα των συναρτήσεων στην απλή προστακτική γλώσσα των διαφανειών. Υποθέστε ότι η γλώσσα υποστηρίζει πραγματικούς αριθμούς, όλες τις αριθμητικές πράξεις και πίνακες. Υποθέστε επίσης ότι όλες οι πράξεις υπολογίζουν πάντα το ιδεατό αποτέλεσμα, χωρίς το ενδεχόμενο υπερχείλισης ή αριθμητικού σφάλματος.

2. Να χρησιμοποιήσετε το εργαλείο επαλήθευσης προγραμμάτων Frama-C, που είναι διαθέσιμο από την ιστοσελίδα <http://frama-c.com/>. Αν το επιχειρήσετε, θα χρειαστεί να εγκαταστήσετε και κάποιο εργαλείο αυτόματης απόδειξης θεωρημάτων, όπως το Alt-Ergo, ή κάποιο σύστημα υποστήριξης αποδείξεων, όπως το Coq. Υπάρχουν σύνδεσμοι προς τέτοια εργαλεία από την ιστοσελίδα του Frama-C.