



Γλώσσες Προγραμματισμού II

Αν δεν αναφέρεται διαφορετικά, οι ασκήσεις πρέπει να παραδίδονται 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 6 Παραλληλισμός και ταυτοχρονισμός στη Haskell

Προθεσμία παράδοσης: 2/2/2020

Σε αυτή την άσκηση σας ζητείται να πειραματιστείτε με τον *παραλληλισμό* και τον *ταυτοχρονισμό* στη Haskell. Ο σκοπός είναι το πρόγραμμά σας να επιτυγχάνει κάποια βελτίωση της επίδοσής του, σε σχέση με αυτή της σειριακής, όταν εκτελείται σε ένα περιβάλλον πολυπύρηνης αρχιτεκτονικής.

Περιγραφή του προβλήματος. Με πόσους τρόπους μπορούμε να επιλέξουμε K στοιχεία από ένα σύνολο N στοιχείων, αν δε μας ενδιαφέρει η σειρά τους; Προφανώς με

$$C(n, k) = \frac{n!}{k!(n-k)!}$$

Ο αριθμός αυτός μπορεί να είναι πολύ μεγάλος, και γι' αυτό θέλουμε να βρούμε το υπόλοιπο (modulo) της διαίρεσής του με κάποιο μεγάλο πρώτο αριθμό P .

Είσοδος και έξοδος. Το πρόγραμμά σας θα διαβάζει τα δεδομένα από την τυπική είσοδο (stdin) και θα τυπώνει τα αποτελέσματα στην τυπική έξοδο (stdout).

Η πρώτη γραμμή της εισόδου περιέχει έναν ακέραιο αριθμό T , το πλήθος των ερωτημάτων που ακολουθούν. Κάθε μία από τις επόμενες T γραμμές περιέχει τρεις ακέραιους αριθμούς, N , K και P , χωρισμένους ανά δύο με ένα κενό διάστημα. Οι αριθμοί αυτοί αναπαριστούν ένα ερώτημα που πρέπει να απαντήσετε. Θεωρήστε ότι $0 \leq K \leq N < P \leq 10^9$ και ότι ο P είναι πρώτος.

Η έξοδος πρέπει να αποτελείται από ακριβώς T γραμμές, κάθε μία από τις οποίες θα περιέχει ακριβώς έναν ακέραιο αριθμό: την απάντηση στο αντίστοιχο ερώτημα.

Τι πρέπει να παραδώσετε.

- Ένα σειριακό πρόγραμμα Haskell, που θα χρησιμοποιήσετε ως βάση για τη σύγκριση της επίδοσης.
- Ένα (ή περισσότερα) παράλληλα ή ταυτόχρονα προγράμματα Haskell, βασισμένα (εναλλακτικά) σε κάποια από τα παρακάτω ή σε συνδυασμούς αυτών:
 - Παραλληλισμός: (a1) Evaluation strategies, (a2) Par monad.
 - Ταυτοχρονισμός: (b1) IO monad και MVar, (b2) Software Transactional Memory.

Η είσοδος και η έξοδος των παράλληλων ή ταυτόχρονων προγραμμάτων θα πρέπει να είναι της ίδιας μορφής με εκείνη του σειριακού.

- Μία σύντομη αναφορά που να περιγράφει τις εμπειρίες και τα συμπεράσματά σας. Η αναφορά θα πρέπει να εξηγήει τους τρόπους “παραλληλοποίησης” που δοκιμάσατε, να παρουσιάζει σε διαγράμματα και να αναλύει τη βελτίωση επίδοσης σε σχέση με το σειριακό πρόγραμμα (scalability) την οποία πετύχατε.