

Ημερομηνία Ανάρτησης: 10/12/2018
Ημερομηνία Παράδοσης: 7/1/2019, 23:59μμ
Αρχές Γλωσσών Προγραμματισμού

1. (10%) Ορίστε σε Haskell την άπειρη λίστα `multiples` η οποία περιέχει όλους τους φυσικούς αριθμούς της μορφής $2^n 3^m 5^k$, $n, m, k \geq 0$ σε αύξουσα σειρά και χωρίς επαναλήψεις, δηλαδή `multiples = [1,2,3,4,5,6,8,9,10,12,15,...]`.
2. (10%) Ορίστε σε Haskell τη συνάρτηση `sssum f a b c` η οποία δεδομένης μιας συνάρτησης `f` με τύπο `Int->Int->Int->Int`, και τριών φυσικών αριθμών `a`, `b`, `c`, υπολογίζει την παράσταση

$$\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^c f \ i \ j \ k$$

Η υλοποίησή σας θα πρέπει να γίνει χωρίς τη χρήση `list comprehensions`.

3. (25%) Ένας μη κατευθυνόμενος γράφος μπορεί να αναπαρασταθεί στη Haskell με ένα ζεύγος (`vs, es`) όπου `vs` είναι μια λίστα φυσικών αριθμών που είναι οι κορυφές του γραφήματος και `es` είναι μια λίστα από ζεύγη φυσικών αριθμών που ανήκουν στη λίστα `vs` και που αναπαριστούν τις πλευρές του γραφήματος. Να γραφεί συνάρτηση `components` σε Haskell η οποία δεδομένου ενός γράφου, μας επιστρέφει ένα ζεύγος του οποίου το πρώτο στοιχείο είναι ο αριθμός των συνεκτικών συνιστωσών του γραφήματος και το δεύτερο στοιχείο είναι μια λίστα που περιέχει σε αύξουσα σειρά τον αριθμό των κορυφών των διαφορετικών συνιστωσών. Για παράδειγμα, `components ([1,2,3,4], [(1,2), (2,3), (3,4), (4,1)])` επιστρέφει `(1, [4])` ενώ `components ([1,2,3,4], [(1,2)])` επιστρέφει `(3, [1,1,2])`.
4. (25%) Ο μικρός Μήτσος μαθαίνει κλάσματα στο σχολείο. Του αρέσουν ιδιαίτερα τα κλάσματα της μορφής $\frac{1}{n}$ γιατί είναι προφανώς πιο απλά. Ρωτάει λοιπόν τη δασκάλα αν κάθε κλάσμα μπορεί να γραφεί ως το άθροισμα ενός αριθμού από απλά κλάσματα τα οποία είναι όλα διαφορετικά μεταξύ τους. Έντρομη η δασκάλα λέει ότι αυτή είναι μια πολύ καλή ερώτηση και να την φέρουν όλα τα παιδιά την επόμενη μέρα λυμένη στο τετράδιο τους. Βοηθήστε τους γονείς των παιδιών να βγάλουν άκρη. Ορίστε σε Haskell τη συνάρτηση `fractions` η οποία δεδομένων δύο αριθμών `p` και `q` επιστρέφει μια λίστα διαφορετικών αριθμών $[n_1, \dots, n_r]$ τέτοια ώστε $\frac{p}{q} = \frac{1}{n_1} + \dots + \frac{1}{n_r}$. Για παράδειγμα, `fractions 4 5` μπορεί να επιστρέφει `[2,4,20]`.
5. (30%) Υλοποιήστε το 3ο ερώτημα της πρώτης εργασίας σε Haskell. Ορίστε συνάρτηση `ugliness l m` η οποία δεδομένων λιστών `l` και `m` μας επιστρέφει το `ugliness` όπως ορίζεται στην εκφώνηση της πρώτης εργασίας. Για παράδειγμα, `ugliness [7,6,1,2,10] [9,11,6,3,12]` επιστρέφει 4.

Παράδοση Ασκήσεων: Η παράδοση πρέπει να γίνει μέχρι τις 23:59μμ, την 7/1/2019. Θα δημιουργήσετε ένα αρχείο το οποίο θα περιέχει τις λύσεις όλων των ασκήσεων και θα το στείλετε με *email* και στις δύο παρακάτω διευθύνσεις: `cs1180004@di.uoa.gr` και `prondo@di.uoa.gr`. **Δεν θα υπάρξει παράταση στην παράδοση των ασκήσεων.** Τα ονόματα των συναρτήσεων που θα χρησιμοποιήσετε στα προγράμματά σας πρέπει να είναι **ακριβώς** τα ίδια με αυτά που καθορίζονται από την παραπάνω εκφώνηση. Καθυστερημένες ασκήσεις δεν θα βαθμολογηθούν.

Σημείωση: Για να μπορέσει κάποιος να λάβει μέρος στην τελική εξέταση του μαθήματος, θα πρέπει να έχει παραδώσει τις δύο πρώτες εργασίες (Prolog και Haskell) με προβιβασμό βαθμό.