ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Εαρινό 2011

ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι

Άσκηση 1

Καταληκτική ημερομηνία και ώρα ηλεκτρονικής υποβολής: 13/5/2011, 12:30

Σκέτη μαγεία (0.25 + 0.25 = 0.5 βαθμοί)

Πάρτε έναν τριψήφιο αριθμό που δεν είναι όλα τα ψηφία του ίδια, έστω τον 846. Ταξινομήστε τα ψηφία του σε φθίνουσα και σε αύξουσα σειρά: 864 και 468. Αφαιρέστε τους δύο αριθμούς: 864−468=396. Επαναλάβετε το ίδιο: 963−369=594. Ξανά: 954−459=495. Ξανά: 954−459=495. Φτάσαμε σε ένα σημείο που, από δω και πέρα, θα επαναλαμβάνεται η ίδια αφαίρεση και θα προκύπτει συνέχεια ο αριθμός 495. Εδώ σταματάμε.

Πάρτε έναν άλλο τριψήφιο αριθμό, έστω τον 452. Κάνετε το ίδιο: 542−245=297, 972−279=693, 963−369=594, 954−459=495. Τυχαίο; Δε νομίζω! Από όποιον αριθμό και αν ξεκινήσετε, σύντομα θα καταλήξετε στο «μαγικό» αριθμό 495.

Γενικεύστε τώρα αυτό για αριθμούς με Ν ψηφία στο σύστημα αρίθμησης με βάση το Β. Πάρτε για παράδειγμα Ν=6 και Β=4. Ένας εξαψήφιος αριθμός στο τετραδικό σύστημα είναι ο $230021_{(4)}$. Η ίδια διαδικασία (στο όμως τετραδικό σύστημα αρίθμησης) μας δίνει: $322100_{(4)}-001223_{(4)}=320211_{(4)}$, $322110_{(4)}-011223_{(4)}=310221_{(4)}$, $322110_{(4)}-011223_{(4)}=310221_{(4)}$. Φθάσαμε επομένως στο «μαγικό» αριθμό $310221_{(4)}=3369$. Για Ν=6 και Β=4, όμως, υπάρχουν και άλλοι δύο αριθμοί στους οποίους μπορεί να καταλήξουμε, εκτός από τον 3369: ο 2550 και ο 3873.

Εν ολίγοις, «μαγικούς» αριθμούς με Ν ψηφία στο σύστημα αρίθμησης με βάση Β ονομάζουμε τους αριθμούς εκείνους που, όταν ταξινομήσουμε τα ψηφία τους σε φθίνουσα σειρά και σε αύξουσα σειρά και αφαιρέσουμε, προκύπτει ο εαυτός τους. Η αντίστοιχη σελίδα του Plus Magazine (http://plus.maths.org/content/mysterious-number-6174) και όλοι οι σύνδεσμοι από εκεί έχουν περισσότερες πληροφορίες για τους «μαγικούς» αριθμούς.

Αυτό που ζητάει η άσκηση είναι να γραφούν δύο προγράμματα (ένα σε C και ένα σε ML) τα οποία να παίρνουν ως είσοδο τα **B** και **N** και να βρίσκουν ένα «μαγικό» αριθμό με Ν ψηφία στο σύστημα αρίθμησης με βάση Β. Προσέξτε ότι αν δεν υπάρχει τέτοιος μαγικός αριθμός, θα πρέπει να επιστρέφεται μηδέν. Στην ιστοσελίδα του μαθήματος υπάρχει ένας σκελετός προγράμματος τον οποίο μπορείτε να χρησιμοποιήσετε για το πρόγραμμά σας σε C. Παρακάτω δείχνουμε κάποιες πιθανές κλήσεις των προγραμμάτων σε C και σε ML.

C

```
> magic 10 3
495
> magic 4 6
3369
> magic 3 4
0
```

ML

```
- magic 10 3;
val it = 495 : int
- magic 4 6;
val it = 3369 : int
- magic 3 4;
val it = 0 : int
```

Μπορείτε να υποθέσετε ότι το **N** είναι το πολύ 64 και το **B** το πολύ 20.

Λόγγοι, βουνά, ραχούλες... (0.25 + 0.25 = 0.5 βαθμοί)

Τοπωνύμιο είναι το όνομα μίας πόλης, ενός χωριού, ενός λόγγου, ενός βουνού, κ.λπ. Πολλές φορές συναντά κανείς τοπωνύμια που μοιάζουν μεταξύ τους πολύ. Για παράδειγμα «Αγγελόκαστρο Κορινθίας» και «Αγγελόκαστρο Αιτωλοακαρνανίας», ή «Γεροπόταμος» και «Γεροπλάτανος».

Τα τοπωνύμια τα αναπαριστούμε με συμβολοσειρές που αποτελούνται από μικρά και κεφαλαία γράμματα του λατινικού αλφαβήτου (A, B, C, ..., Z, a, b, c, ..., z) και από κενά διαστήματα. Όμως, δεν μπορεί να εμφανίζονται δύο ή περισσότερα διαδοχικά κενά διαστήματα και δεν υπάρχουν κενά διαστήματα στην αρχή ή στο τέλος των τοπωνυμίων. Οι συμβολοσειρές που αποτελούνται από τους *m* πρώτους χαρακτήρες ενός τοπωνυμίου ονομάζονται «προθέματα» μήκους *m*. Π.χ. η συμβολοσειρά «Γεροπ» είναι ένα πρόθεμα μήκους 5 του τοπωνυμίου «Γεροπόταμος».

Ο βαθμός ομοιότητας Ls(*T*) ενός συνόλου τοπωνυμίων *T* είναι το μήκος του μακρύτερου κοινού προθέματος όλων των τοπωνυμίων του *T*. Για παράδειγμα, για το σύνολο *T* = {«Γεροπλάτανος», «Γεροπόταμος», «Γεροβασιλιώτικα»}, ο βαθμός ομοιότητας είναι Ls(*T*) = 4 γιατί το μακρύτερο κοινό πρόθεμα είναι «Γερο».

Ο βαθμός πολυπλοκότητας Lc(*T*) ενός συνόλου *T* αποτελούμενου από *k* τοπωνύμια ορίζεται ως

$$\mathrm{Lc}(T) = k \times \mathrm{Ls}(T)$$

Για παράδειγμα, για το σύνολο *T* = {«Γεροπλάτανος», «Γεροπόταμος», «Γεροβασιλιώτικα»}, ο βαθμός πολυπλοκότητας είναι Lc(*T*) = 3×4 = 12.

Γράψτε προγράμματα (ένα σε C και ένα σε ML) που για κάποιο σύνολο τοπωνυμίων S βρίσκει το υποσύνολο $T \subseteq S$ με το μέγιστο βαθμό πολυπλοκότητας και να επιστρέφει αυτόν τον αριθμό.

Η είσοδος πρέπει να διαβάζεται από ένα αρχείο το οποίο έχει την εξής μορφή: η πρώτη γραμμή του περιέχει το πλήθος των τοπωνυμίων *N*. Κάθε μία από τις επόμενες *N* γραμμές περιέχει ένα τοπωνύμιο, δηλαδή μία συμβολοσειρά αποτελούμενη από γράμματα και κενά διαστήματα, με τους περιορισμούς που αναφέρθηκαν για τη χρήση των κενών διαστημάτων. Το πρόγραμμα θα πρέπει να τυπώνει (στη C) ή να επιστρέφει (στην ML) έναν ακέραιο αριθμό, το μέγιστο βαθμό πολυπλοκότητας που μπορεί να έχει κάποιο υποσύνολο των τοπωνυμίων.

Έστω ότι το αρχείο **`input.txt`** έχει το παρακάτω περιεχόμενο:

```
7
Katsikochori
Geroplatanos
Geropotamos
Katsiantoneika
Katsikorachoula
Gerovassiliotika
Katsigianneika
```

Σε ένα σύστημα Unix, η χρήση του προγράμματος σε C δείχνει ως εξής:

```
> gcc -Wall -O3 -o site_names  yourfile.c
> ./site_names < input.txt
20
```

Η χρήση του προγράμματος σε ML δείχνει ως εξής:

```
- site_names "input.txt";
val it = 20 : int
```

Η συγκεκριμένη άσκηση ζητάει η λύση σας να είναι αποδοτική. Για το σκοπό αυτό θα πρέπει να χρησιμοποιήσετε μια δομή δεδομένων που να ταιριάζει στο συγκεκριμένο πρόβλημα. Μια τέτοια δομή είναι τα *tries* (http://en.wikipedia.org/wiki/Trie) αλλά ίσως υπάρχουν και άλλες κατάλληλες δομές. Για την υλοποίηση αυτών, μπορείτε αν θέλετε να χρησιμοποιήσετε κώδικα που θα βρείτε κάπου στο διαδίκτυο, αλλά στην περίπτωση αυτή η προέλευσή του πρέπει να αναφέρεται στα σχόλια του προγράμματός σας. Το υπόλοιπο πρόγραμμα πρέπει να είναι δικό σας.

Περαιτέρω οδηγίες για την άσκηση

- Μπορείτε να δουλέψετε σε ομάδες το πολύ 2 ατόμων.
- Δεν επιτρέπεται να μοιράζεστε ασκήσεις με άλλους συμφοιτητές σας ή να βάλετε τις ασκήσεις σας σε μέρος που άλλοι μπορούν να τις βρουν (π.χ. σε κάποια σελίδα στο διαδίκτυο, σε ιστοτόπους συζητήσεων, …).
- Τα προγράμματα σε C πρέπει να είναι σε ένα αρχείο και να μπορούν να μεταγλωττιστούν χωρίς warnings με gcc με μια εντολή της μορφής: **gcc -Wall -O3 -lm -o file file.c**
- Τα προγράμματα σε ML πρέπει επίσης να είναι σε ένα αρχείο και να δουλεύουν σε SML/NJ (v110.67 ή πιο πρόσφατη έκδοση) ή σε Mlton (20070826 ή πιο πρόσφατη). Το σύστημα ηλεκτρονικής υποβολής σας επιτρέπει να επιλέξετε μεταξύ αυτών των υλοποιήσεων της ML.
- Η αποστολή των προγραμμάτων θα γίνει ηλεκτρονικά μέσω του moodle. Θα υπάρξει σχετική ανακοίνωση για την ακριβή διαδικασία υποβολής. Τα προγράμματά σας πρέπει να διαβάζουν την είσοδο όπως αναφέρεται και δεν πρέπει να έχουν κάποιου άλλους είδους έξοδο διότι δε θα γίνουν δεκτά από το σύστημα στο οποίο θα υποβληθούν. Για τη διευκόλυνσή σας, στο site του μαθήματος υπάρχει ένα αρχείο με το σκελετό του προγράμματος για το **magic** σε C (**magic.c**) καθώς και μια συνάρτηση σε ML (στο αρχείο **site_names_parse.sml**) που μπορείτε να χρησιμοποιήσετε για να διαβάσετε την είσοδο της δεύτερης άσκησης από ένα αρχείο εισόδου. Βεβαίως η χρησιμοποίηση των παραπάνω στα προγράμματά σας είναι προαιρετική.