

ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ: 2^Η ΑΣΚΗΣΗ- ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΣΦΑΤΩΝ ΚΛΗΣΕΩΝ ΚΙΝΗΤΟΥ

Το αντικείμενο της 2^{ης} άσκησης είναι η διαχείριση του μητρώου προσφάτων κλήσεων κινητού τηλεφώνου. Το κινητό διαχειρίζεται 3 κατηγορίες προσφάτων κλήσεων, ως λίστες : Αναπάντητες, Ληφθείσες και Κληθέντες αριθμοί, που αναφέρουν τον αριθμό (ή το όνομα της επαφής) και είναι ταξινομημένες βάσει ημερομηνίας και ώρας κλήσης. Ο χρήστης μπορεί να ορίζει τον μέγιστο αριθμό κλήσεων που αποθηκεύονται σε κάθε λίστα.

1. Περιγραφή του προβλήματος. Τα στοιχεία κλήσεων δίνονται σε txt αρχείο κλήσεων, που κάθε γραμμή του έχει μια από τις 4 μορφές (σε μορφή σχολίων η σημασιολογία της). Σας δίδεται το Calls.txt.

```
M <int> //Ο μέγιστος αριθμός int κλήσεων στις λίστες
R dd/mm/yyyy time(sec) αριθμός // κλήση ημέρα-ώρα-αριθμός στις Ληφθείσες
U dd/mm/yyyy time(sec) αριθμός // κλήση ημέρα-ώρα-αριθμός στις Αναπάντητες
C dd/mm/yyyy time(sec) αριθμός // κλήση ημέρα-ώρα-αριθμός στους Κληθέντες αριθ.
```

Η διάταξη των στοιχείων στο αρχείο είναι από την παλαιότερη προς την πλέον πρόσφατη. Για απλούστευση η ώρα κλήσης (time) απεικονίζεται με το δευτερόλεπτο έναρξης κλήσης (από 0 έως 24Χ60Χ60) μέσα στην ημέρα. Ο αριθμός τηλεφώνου είναι string [15]. Τα στοιχεία αυτά στην πραγματικότητα παράγονται αυτόματα από άλλο σύστημα (εκτός ενδιαφέροντος της άσκησης) και τα θεωρούμε σωστά, δεν χρειάζονται δηλαδή κανέναν έλεγχο ορθότητας (ημερομηνίες, ώρες, η διάταξη τους και οι αριθμοί τηλεφώνων σωστά). Οι γραμμές M <int> απεικονίζουν την εντολή του χρήστη που ορίζει τον μέγιστο αριθμό κλήσεων σε κάθε λίστα. Αν οι λίστες είναι μεγαλύτερες, τα απαραίτητα παλαιότερα στοιχεία αφαιρούνται και στην συνέχεια το μέγεθός των λιστών δεν πρέπει να υπερβαίνει τον αριθμό αυτόν.

Επίσης ο χρήστης μπορεί να διαγράψει εντελώς μια κλήση ή αλλάξει τον αριθμό τηλεφώνου σε όλες τις λίστες (στην πραγματικότητα γίνεται μόνο αν αλλάξει ο αριθμός μιας επαφής). Τα στοιχεία των διαγραφών ή αλλαγών δίνονται σε txt αρχείο Αλλαγών που κάθε γραμμή του έχει μια από τις 2 μορφές (σε μορφή σχολίων η σημασιολογία της).

Δίδεται το DelOrChNumber.txt.

```
A tel1 tel2 // αλλαγή τηλεφώνου όπου εμφανίζεται το tel1 σε tel2
D tel // διαγραφή κλήσης όπου εμφανίζεται το tel
```

Τέλος θέλουμε να πειραματιστούμε με μια νέα λειτουργία, που είναι η αναζήτηση της πιο πρόσφατης κλήσης σε κάποια ορισμένη ημερομηνία. Τα στοιχεία των αναζητήσεων δίνονται σε txt αρχείο Αναζητήσεων που κάθε γραμμή του έχει μια από τις 3 μορφές (σε μορφή σχολίων η σημασιολογία της). Δίδεται το SearchCalls.txt.

```
R dd/mm/yyyy // Αναζήτηση κλήσης βάσει ημερομηνίας στις Ληφθείσες
U dd/mm/yyyy // Αναζήτηση κλήσης βάσει ημερομηνίας στις Αναπάντητες
C dd/mm/yyyy // Αναζήτηση κλήσης βάσει ημερομηνίας στους Κληθέντες Αριθμούς
```

2. Σχεδιασμός και Δομή του Προγράμματος Σας δίδεται σκελετός προγράμματος και αρχεία δεδομένων.

2.1 Η Υλοποίηση βασίζεται στον ΑΤΔ OuraPlus, που ενσωματώνει βασικές πράξεις της ουράς και της λίστας. Η διεπαφή της ΑΤΔ OuraPlus με Ολική Απόκρυψη είναι:

```
typedef struct GeneralNode *TOuraPlus; // ο τύπος της
```

```
void OuraPlus_dimiourgia(TOuraPlus * const ouraP);
int OuraPlus_keni(const TOuraPlus ouraP);
int OuraPlus_megethos(const TOuraPlus ouraP);
int OuraPlus_eisagogiTelos(const TOuraPlus ouraP, TStoixeiouOuraPlus stoixeio);
int OuraPlus_periexomenoTrexon(const TOuraPlus ouraP, TStoixeiouOuraPlus *const value);
int OuraPlus_prwtos(const TOuraPlus ouraP);
int OuraPlus_epomenos(const TOuraPlus ouraP);
int OuraPlus_teleytaios(const TOuraPlus ouraP);
int OuraPlus_proigoymenos(const TOuraPlus ouraP);
int OuraPlus_exagogiArxh(const TOuraPlus ouraP);
int OuraPlus_diagrafiTrexon(const TOuraPlus ouraP);
int OuraPlus_allagiTrexon(const TOuraPlus ouraP, TStoixeiouOuraPlus stoixeio);
int OuraPlus_anazitisi(const TOuraPlus ouraP, TStoixeiouOuraPlus stoixeio,
                      int (*iso)(TStoixeiouOuraPlus ,TStoixeiouOuraPlus));
void OuraPlus_katastrofi(TOuraPlus *const ouraP);
```

Οι ανωτέρω πράξεις δίνονται με την σειρά που προτείνεται να τις αναπτύξετε. Μετά από κάποια απαραίτητη πράξη ακολουθούν οι πιο εύκολες (πχ. μετά την OuraPlus_dimiourgia ακολουθούν οι OuraPlus_keni και OuraPlus_megethos. Μετά την OuraPlus_eisagogiTelos και OuraPlus_periexomenoTrexon ακολουθούν οι OuraPlus_prwtos, OuraPlus_epomenos, OuraPlus_teleytaios και OuraPlus_proigoymenos, κλπ)

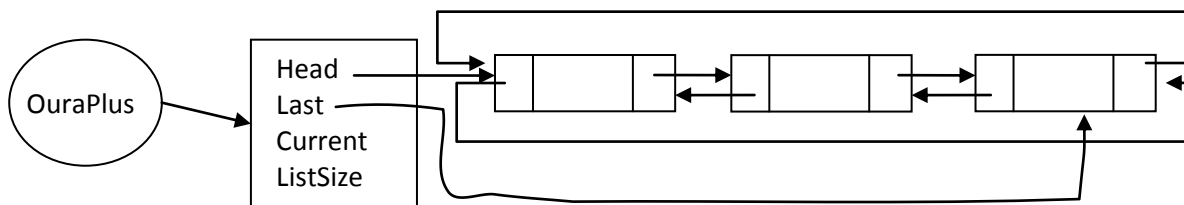
Η υλοποίηση της OuraPlus να γίνει με **Κυκλική Διπλά Συνδεδεμένη Λίστα με δείκτες** στον πρώτο, τελευταίο και τρέχοντα (βοηθητικός δείκτης) κόμβο. Επίσης να συντηρεί το μέγεθος της λίστας. (Για βοήθεια μπορείτε να δείτε τις υλοποιήσεις 7.5 και 7.6 της λίστας). Ο σχεδιασμός υλοποιείται με την εξής δομή:

```
typedef struct typos_komvou *typos_deikti;

typedef struct typos_komvou{
    TStoixeiouOuraPlus dedomena; /*ta dedomena*/
    typos_deikti epomenos; /*deikths ston epomeno komvo*/
    typos_deikti proigoumenos; /*deikths ston proigoumeno komvo*/
} typos_komvou;

typedef struct GeneralNode{
    typos_deikti Head, /* h arxh ths listas */
    Last, /* to telos ths listas */
    Current; /* o trexon komvos ths listas */
    int ListSize; /* to megthethos ths listas */
}GeneralNode;
```

Η γενική δομή (π.χ. λίστα τριών κόμβων) που θέλουμε να συντηρήσουμε έχει την εξής μορφή:



Προτείνεται να αναπτύξετε και να δοκιμάσετε την υλοποίηση του ΑΤΔ OuraPlus με το `TestOuraPlus.c` με τον τύπο στοιχείου `int`, που σας έχει δοθεί. Δοκιμάστε μια-μια τις πράξεις καθώς τις αναπτύσσετε.

2.2. Ο τύπος στοιχείου ορίζει την εξής διεπαφή (σας δίδεται ως `int` για τις δοκιμές της OuraPlus μόνο) , με Μερική Απόκρυψη. Δεν ζητείται και δεν είναι απαραίτητη η Ολική Απόκρυψη του Τύπου Στοιχείου..

```
typedef struct {...} TStoixeiouOuraPlus; /* να την ολοκληρώσετε με date, sec, number */
```

```
int TSOuraPlus_setValue (TStoixeiouOuraPlus *target, TStoixeiouOuraPlus source);
int TSOuraPlus_readValue (FILE *from, TStoixeiouOuraPlus *Elem);
int TSOuraPlus_writeValue(FILE *to, TStoixeiouOuraPlus Elem);
int TSOuraPlus_iso(TStoixeiouOuraPlus Elem1, TStoixeiouOuraPlus Elem2);
```

Να ορίσετε τον τύπο στοιχείου. Οι `TSOuraPlus_setValue`, `TSOuraPlus_readValue`, `TSOuraPlus_writeValue` υλοποιούνται όπως στην πρώτη άσκηση με την επιπλέον δυνατότητα να επιστρέφουν κωδικό λάθους. Η συνάρτηση `TSOuraPlus_iso` δέχεται δυο στοιχεία κλήσεων και συγκρίνει **μόνο** τις ημερομηνίες τους για ισότητα. Επιστρέφει 1 αν είναι ίσες, αλλιώς 0. Υποθέστε ότι οι ημερομηνίες είναι σωστές. Προτείνεται να δοκιμάσετε ξεχωριστά τις υλοποιήσεις του τύπου στοιχείου με το `TestTypoStoixeiou.c` που σας δίδεται . Κατόπιν δοκιμάστε πάλι την υλοποίηση του ΑΤΔ OuraPlus με τον νέο τύπο στοιχείου με το πρόγραμμα `TestOuraPlus.c`, όπως στο 2.1.

2.3 Η Διαχείριση των λιστών προσφάτων κλήσεων γίνεται στο κυρίως πρόγραμμα `MobileCallManagement.c`, το οποίο πρέπει να ολοκληρώσετε. Ειδικότερα να αναπτύξετε τις πιο κάτω συναρτήσεις (6), οι οποίες καλούν πράξεις του ΑΤΔ OuraPlus και `TypoStoixeiou`.

```
int Initialize(); //Αρχικοποιεί τις 3 OuraPlus. Καλεί την πράξη OuraPlus_dimiourgia .
```

```
int PopulateLists(FILE *in); // Διαβάζει στοιχεία από το αρχείο Κλήσεων και τα εισάγει στο τέλος της αντίστοιχης OuraPlus. Αν μια λίστα έχει μεγαλώσει περισσότερο από το μέγιστο διαγράφει τον απαραίτητο αριθμό από παλαιότερα στοιχεία (δηλαδή από την αρχή της λίστας). Καλεί τις πράξεις TSOuraPlus_readValue, OuraPlus_keni, OuraPlus_megethos, OuraPlus_eisagogiTelos and OuraPlus_exagogiArxh.
```

```
int ShowList(FILE *out, TOuraPlus CallList, int order); // Διαδρομή, τυπώνει τα στοιχεία της CallList στο out (ένα σε κάθε γραμμή) από τα παλαιότερα προς τα πιο πρόσφατα αν order==1, αντίστροφα αν order==-1; Καλεί τις πράξεις, OuraPlus_prwtos, OuraPlus_periexomenoTrexon, OuraPlus_epomenos, OuraPlus_teleytaios, OuraPlus_proigoymenos, TSOuraPlus_writeValue.
```

```
int DeleteOrChangeCalls(FILE *in); // Διαβάζει στοιχεία από το αρχείο Αλλαγών και με διαδρομή αλλάζει τον αριθμό ή διαγράφει εντελώς τις κλήσεις από/προς τον αριθμό αυτό σε όλες τις εμφανίσεις του. Καλεί τις πράξεις OuraPlus_prwtos, OuraPlus_allagiTrexon, OuraPlus_diagrafiTrexon, OuraPlus_epomenos.
```

```
int SearchCallList(FILE *in, FILE *out); // Διαβάζει στοιχεία από το αρχείο Αναζητήσεων in και εκτυπώνει στο out την κλήση αριθμού που βρήκε. Καλεί τις πράξεις OuraPlus_periexomenoTrexon και OuraPlus_anazitisi, η οποία χρησιμοποιεί ως παράμετρο την TSOuraPlus_iso και την TSOuraPlus_writeValue.
```

```
int Destroy(); // Καταστρέφει τις OuraPlus. Καλεί την OuraPlus_katastrofi.
```

Στα σχόλια αναφέρονται οι πράξεις του OuraPlus και ΤυροΣτοιχείου που απαιτούνται. Έτσι αν κάποιος έχει π.χ. ολοκληρώσει τις πρώτες 4 πράξεις του OuraPlus μπορεί να ολοκληρώσει την Initialize και την PopulateLists.

3. Προθεσμία Υποβολής ΚΥΡΙΑΚΗ 13/05/2012

4. Παραδοτέα

1. Πηγαίος κώδικας (.h και .c αρχεία) Αρχεία δεδομένων, αν λειτουργεί με ειδικά δεδομένα.
2. Αν χρησιμοποιήσατε DevC++ παραδώστε και το project file.
3. Τεκμηρίωση (μέγιστο 1 σελίδα) Σύντομο κείμενο (pdf) με την εξής δομή
 - I. Τα στοιχεία σας: (Όνομα-Επώνυμο-AM)
 - II. Λειτουργικότητα: Να περιγράψετε τι κάνει το πρόγραμμά σας (μπορεί να κάνει περισσότερα ή και λιγότερα από τα ζητούμενα της άσκησης). Ειδικότερα, αν το πρόγραμμά σας δεν υλοποιεί πλήρως την άσκηση να αναφέρετε ακριβώς τις δυνατότητες και τις ελλείψεις του.
 - III. Οδηγίες Χρήσης του προγράμματος σας: π.χ. Διάταξη δεδομένων εισόδου.
 - IV. Περιβάλλον Υλοποίησης και Δοκιμών: πχ. Αναπτύχθηκε σε Dev C++ σε περιβάλλον Windows XP, δοκιμάστηκε επίσης σε gcc και Unix.
4. Συμπληρώστε στο eclass τις ερωτήσεις με τις ώρες απασχόλησής σας, κλπ.

5. Τρόπος Αξιολόγησης

Οι ασκήσεις είναι ατομικές και θα ελεγχθούν για ομοιότητες χρησιμοποιώντας ειδικό σύστημα εντοπισμού ομοιοτήτων. Σε περίπτωση μεγάλης «ομοιότητας» όλες οι «παρόμοιες» ασκήσεις θα μηδενιστούν.

Αξιολογούνται η λειτουργικότητα, η δομή και η τεκμηρίωση του προγράμματος. Για να αξιολογηθεί το πρόγραμμα σας πρέπει τουλάχιστον να μεταγλωττίζεται. Αν δεν μεταγλωττίζεται δεν παίρνει βαθμό. Δεν πρέπει να δίνει segmentation fault, σε αυτήν την περίπτωση αξιολογείται η λειτουργικότητά του μέχρι το σημείο που λειτουργεί. Αν λειτουργεί με δικά σας δεδομένα, να τα συμπεριλάβετε. Πριν παραδώσετε το πρόγραμμά σας δοκιμάστε το μια τελευταία φορά και βεβαιωθείτε ότι παραδίδετε τα σωστά αρχεία. Αναλυτικά:

Λειτουργικότητα (87/100)

OuraPlus_dimiourgia, OuraPlus_eisagogiTelos, OuraPlus_epomenos, OuraPlus_proigoymenos, OuraPlus_exagogiArxh	καθεμία 5	Σύνολο 25
OuraPlus_keni, OuraPlus_megethos, OuraPlus_prwtos, OuraPlus_teleytaios, OuraPlus_periexomenoTrexon, OuraPlus_allagiTrexon	2	12
OuraPlus_diagrafiTrexon, OuraPlus_katastrofi	10	20

TSOuraPlus_setValue, TSOuraPlus_readValue, TSOuraPlus_writeValue	2	6
--	---	---

Initialize, Destroy	2	4
PopulateLists, ShowList, DeleteOrChangeCalls, main	5	20

Δομή (ακολουθείστε την δομή που σας δίδεται) (8/100)

Τεκμηρίωση και Παρουσίαση (5/100)

Συνολικό BONUS για τις SearchCallList, OuraPlus_anazitisi, TSOuraPlus_iso		10
---	--	----

Σύνολο (100 + 10 bonus)