

2η ΑΣΚΗΣΗ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ 2017-18

Σε πολλά εστιατόρια η παραγγελία δίδεται μέσω palmtops. Απαραίτητη λειτουργία του προγράμματος παραγγελιών είναι η διαχείριση του μενού του εστιατορίου. Ζητείται να αναπτύξετε σε C έναν ΑΤΔ Menu καθώς και ένα πρόγραμμα πελάτη που να δημιουργεί και να διαχειρίζεται δυο μενού, ένα εστιατορίου και ένα καφετέρειας. Ακολουθεί ο ορισμός του ΑΤΔ Menu.

Ορισμός ΑΤΔ MENU

Είναι μια Λίστα στοιχείων της οποίας κάθε κόμβος περιέχει δεδομένα, είτε κατηγορία πιάτων είτε πιάτο με την τιμή του. Αν είναι κατηγορία πιάτων μπορεί να έχει υπο-κατηγορία, που είναι λίστα ίδιου τύπου. Το Μενού δηλαδή είναι, Λίστα από Λίστες.

Οι κόμβοι στην ίδια κατηγορία ή υπο-κατηγορία πιάτων συνδέονται σε **διπλή συνδεδεμένη λίστα με κεφαλή** με χρήση δυο δεικτών prev και next. Η κεφαλή δεν περιέχει χρήσιμα δεδομένα, απλυστεύει τις πράξεις εισαγωγή και διαγραφή κόμβων. Επίσης οι κόμβοι έχουν δυο δείκτες up και down. Αν ο κόμβος είναι κατηγορία πιάτων ο δείκτης down δείχνει στον κόμβο κεφαλή της υπο-κατηγορίας του. Αν είναι πιάτο ο down είναι πάντα NULL. Ο δείκτης up των κόμβων σε μια υπο-κατηγορία δείχνει στον κόμβο που ανήκει η υπο-κατηγορία. Ο δείκτης up της ανώτερης κατηγορίας είναι NULL.

Τα δεδομένα κόμβων είναι α) ένα string μεταβλητού μήκους συμβολοσειρών β) ένδειξη ότι πρόκειται για κατηγορία πιάτων ή όχι και γ) την τιμή του πιάτου, σε περίπτωση που είναι πιάτο. Ένας κόμβος μπορεί να είναι είτε κατηγορία φαγητών είτε συγκεκριμένο πιάτο με τιμή.

Οι Τύποι που απαιτούνται (υπάρχουν και στον αρχικό σκελετό προγράμματος που σας δίδεται)

```
typedef struct InfoMenu *InfoMenuPtr; // Ο Τύπος δείκτη σε Πληροφοριακό Κόμβο Μενού
```

```
typedef struct MenuNode *MenuNodePtr; // Ο τύπος δείκτη σε κόμβο
```

```
struct InfoMenu { // ο πληροφοριακός κόμβος
    char*      Name;           // το όνομα του εστιατορίου
    MenuNodePtr Start;         // Δείκτης στον πρώτο κόμβο της ανώτερης κατηγορίας
    MenuNodePtr Current;       // Δείκτης στον τρέχοντα κόμβο
    MenuNodePtr CutToPaste;    // Δείκτης στον κόμβο προς μεταφορά
};
```

```
struct MenuNode { // ο κόμβος για κατηγορία ή πιάτο
    MenuNodePtr Prev, Next;    // οι δείκτες για την διπλά συνδεδεμένη λίστα
    Up, Down;                 // οι δείκτες για κατηγορία-υποκατηγορία
    TStoixeiouMenu Data;
};
```

Ο τύπος στοιχείου και οι πράξεις

```
typedef struct Komvos {
    char * name; // Name of Category or Plate
    int Category; // 0 if Plate, 1 if Category
    float Price; // Valid only if Plate, η τιμή του
} TStoixeiouMenu;
```

```
void TSSetValue(TStoixeiouMenu *Lhs, TStoixeiouMenu Rhs); //Lhs=Rhs
int TSPrintValue(FILE *out, TStoixeiouMenu V);
int TSReadValue(FILE *in, TStoixeiouMenu * V);
```

Για μενού εστιατορίου η δήλωση (στο main)

```
InfoMenuPtr FoodMenu=NULL, // Το μενού του εστιατορίου
CafeMenu=NULL; // Το μενού της καφετέρειας
```

Οι Πράξεις του ΑΤΔ MENU

ΜενούΔημιουργία	Δημιουργεί τον πληροφοριακό κόμβο InfoMenu, αρχικοποιώντας τα μέλη του. Οι δείκτες Start και Current στον κόμβο κεφαλή, ο δείκτης CutToPaste σε NULL, και όνομα εστιατορίου από παράμετρο.
ΜενούΚενήΚατηγορία	Επιστρέφει 1 (κενή) αν ο Current δείχνει σε κεφαλή, η οποία δεν έχει επόμενο κόμβο, αλλιώς 0 (μη κενή).
ΜενούΕισαγωγήΚόμβου	Εισάγει νέο κόμβο (στη ίδια κατηγορία) μετά από τον κόμβο στον οποίο δείχνει ο Current. Ενημέρωση Prev, Next και των δεδομένων των κόμβων. Θέτει τον Down = NULL και τον Up στην ίδια τιμή με το Up του Current. Θέτει τον Current στον νέο κόμβο.
ΜενούΕκτύπωσεΠεριεχόμενο	Εμφανίζει τα δεδομένα του κόμβου που δείχνει ο Current (κατηγορία ή πιάτο με την τιμή του). Σφάλμα, αν ο Current βρίσκεται σε κεφαλή.
ΜενούΑλλαγήΠεριεχόμενου	Αλλάζει το string και/ή τιμή (αν είναι πιάτο) του κόμβου που δείχνει ο current. Αν ο Current βρίσκεται σε κεφαλή, είναι σφάλμα.
ΜενούΕπόμενος	Θέτει τον Current στον επόμενο κόμβο της ίδιας κατηγορίας (ακολουθεί τον Next). Αν είναι στον τελευταίο κόμβο, παραμένει εκεί, δεν γίνεται NULL.
ΜενούΠροηγούμενος	Θέτει τον Current στον προηγούμενο κόμβο της ίδιας κατηγορίας (ακολουθεί τον Prev). Αν είναι σε κόμβο κεφαλή παραμένει εκεί, δεν γίνεται NULL.
ΜενούΜετακίνησηΜετά	Ανταλλαγή θέσεων του κόμβου (όχι κεφαλής) που δείχνει ο Current με τον επόμενο του μαζί με πιθανές υπο-κατηγορίες. Ο Current δεν αλλάζει. Αν δεν υπάρχει επόμενος είναι σφάλμα.
ΜενούΜετακίνησηΠριν	Ανταλλαγή θέσεων του κόμβου που δείχνει ο Current με τον προηγούμενό (όχι κεφαλής) του μαζί με πιθανές υπο-κατηγορίες. Ο Current δεν αλλάζει. Αν δεν υπάρχει υπάρχει προηγούμενος είναι σφάλμα.
ΜενούΕισαγωγήΚατηγορία	Εισάγει νέο κόμβο κεφαλή νέας υπο-κατηγορίας στο Down του κόμβου στον οποίο δείχνει ο Current. Αν ο Current βρίσκεται σε κεφαλή, είναι σφάλμα. Ο Up του κόμβου κεφαλής δείχνει τον κόμβο της άνω κατηγορίας (Current) και Down, Prev, Next =NULL. Ο Current παραμένει στον κόμβο.
ΜενούΜετακίνησηΠάνω	Θέτει τον Current στον κόμβο της άνω κατηγορίας, αν υπάρχει, αλλιώς παραμένει εκεί.
ΜενούΜετακίνησηΚάτω	Θέτει τον Current στην κεφαλή της υπο-κατηγορίας ενός κόμβου (down), αν υπάρχει, αλλιώς παραμένει εκεί.
ΜενούΕκτύπωσεΚατηγορία	Δέχεται δείκτη στην αρχή κατηγορίας (ο Current σε κόμβο κεφαλή) και εκτυπώνει το περιεχόμενο όλων των κόμβων της κατηγορίας και όλων των υπο-κατηγοριών. Δεν εκτυπώνει κόμβους κεφαλή.
ΜενούΔιαγραφή	Διαγράφει τον κόμβο μετά τον κόμβο που δείχνει ο Current (αφορά Prev+Next). Ο Current παραμένει. Αν ο κόμβος που πρόκειται να διαγραφεί έχει υπο-υποκατηγορία να μην εκτελείται η πράξη. Αν ο Current δεν έχει επόμενο επίσης να μην εκτελείται.
ΜενούΔιαγραφήΚάτω	Διαγράφει όλες τις υπο κατηγορίες κόμβων με αρχή τον δείκτη Down του κόμβου που δείχνει ο Current. Αν οι υποκατηγορίες έχουν και αυτές υπο-κατηγορίες διαγράφονται επίσης. Ο Current παραμένει. (σημ. οι κόμβοι κεφαλή είναι οι μοναδικοί που έχουν Prev==NULL).
ΜενούΑποκοπή (Cut)	Αποσπά τον κόμβο από την λίστα που δείχνει ο Current, τον διαγράφει από την λίστα αλλά αντί να τον ελευθερώνει (free), ενημερώνει τον δείκτη CutToPaste. Ο Current πηγαίνει στον προηγούμενο κόμβο.
ΜενούΕπικόλληση (Paste)	Επικολλά τον αποσπασμένο Κόμβο (δείκτης CutToPaste) μετά από τον Current (στην λίστα prev-next). Ο Current πηγαίνει σε αυτόν.
ΜενούΚαταστροφή	Ελευθερώνει όλη την μνήμη του μενού, κατηγορίες, πιάτα, πληροφοριακό κόμβο
Bonus	Input/Output μενού από/σε html αρχείο. έχει δοθεί input αρχείο.

Οδηγίες Σχεδίασης και Ανάπτυξης Προγράμματος

Το πρόγραμμά σας πρέπει να είναι οργανωμένο σε ενότητες (modules) και σε πρόγραμμα-πελάτη που θα καλεί σε case-switch όλες τις συναρτήσεις του Menu.h, όπως τα προγράμματα δοκιμών για στοίβα και ουρά. Το πρόγραμμα πελάτη να μην χρησιμοποιεί άμεσα την δομή των ενοτήτων στα .h παρά μόνο μέσω των συναρτήσεων που ορίζονται. Για τον ΑΤΔ Μενού να χρησιμοποιήσετε αποκλειστικά Ολική Απόκρυψη. Σας δίδεται αρχικό πρόγραμμα που θα επεκτείνετε. Επίσης να ολοκληρώσετε τις συναρτήσεις στο Τύπο Στοιχείου. Να ολοκληρώσετε την υλοποίηση με βάση τις αρχές των Ενοτήτων (Διαφάνειες Modularity Πέμπτης 29/3).

Συνιστάται να κρατάτε αντίγραφα των προγραμμάτων σε κάθε στάδιο ανάπτυξης. Ένα αντίγραφο θα σας φανεί πολύ χρήσιμο αν θέλετε να επιστρέψετε σε προηγούμενη έκδοση.

Παραδοτέα

1. Πηγαίος κώδικας (όλα τα .c, .h αρχεία σας). Επίσης το Makefile για gcc-linux. Σε κάθε αρχείο να αναφέρετε το όνομά σας στο εισαγωγικό σχόλιο.
2. Τεκμηρίωση (μέγιστο 1 σελίδα) Σύντομο κείμενο (pdf) με την εξής δομή
 - Τα στοιχεία σας: (Όνομα-Επώνυμο-ΑΜ)
 - Λειτουργικότητα: Να περιγράψετε ποιες συναρτήσεις υλοποιεί το πρόγραμμά σας (μπορεί να κάνει περισσότερα ή και λιγότερα από τα ζητούμενα της άσκησης).
 - Οδηγίες Χρήσης του προγράμματος σας: π.χ. Διάταξη δεδομένων εισόδου.
 - Περιβάλλον Υλοποίησης και Δοκιμών: πχ. Αναπτύχθηκε σε gcc σε Linux

Οδηγίες Παράδοσης

Το αρχείο τεκμηρίωσης μαζί με τα αρχεία του προγράμματος να τα βάλετε σε έναν φάκελο, τον οποίο θα συμπίεσετε (zip, rar) με όνομα τον ΑΜ σας και θα ανεβάσετε στο eclass.

Τρόπος Αξιολόγησης

Οι ασκήσεις είναι **ατομικές** και θα ελεγχθούν για ομοιότητες χρησιμοποιώντας ειδικό σύστημα εντοπισμού ομοιοτήτων/αντιγραφών. Σε περίπτωση μεγάλης «ομοιότητας» όλες οι «παρόμοιες» ασκήσεις θα θεωρούνται αντιγραφές και όλοι οι εμπλεκόμενοι θα κόβονται στο μάθημα για το τρέχον ακαδημαϊκό έτος.

Θα αξιολογηθούν η λειτουργικότητα, η δομή και η τεκμηρίωση του προγράμματος. Αναλυτικά:

Λειτουργικότητα (85/100)

Το main και κάθε συνάρτηση (18) από 5 βαθμούς, εκτός από τα ζεύγη Επόμενος/Προηγούμενος και Πάνω/Κάτω που παίρνουν μαζί από 5 βαθμούς.

Δομή (10/100)

Οργάνωση σε Ενότητες (.h, .c) και πρόγραμμα πελάτη	(05)
Απόκρυψη Υλοποίησης, σωστή χρήση στο πρόγραμμα-πελάτη	(05)

Τεκμηρίωση και Παρουσίαση (5/100)

ΠΡΟΣΟΧΗ:

Για να αξιολογηθεί το πρόγραμμά σας (έστω για την δομή του) πρέπει τουλάχιστον να μεταγλωττίζεται. Αν δεν μεταγλωττίζεται δεν παίρνει βαθμό. Πριν παραδώσετε το πρόγραμμά σας δοκιμάστε το μια τελευταία φορά και βεβαιωθείτε ότι παραδίδετε τα σωστά αρχεία.

Παράδειγμα μενού εστιατορίου

- Μενού Εστιατορίου ΚΑΛΟ-ΦΤΗΝΟ-και-ΜΠΟΛΙΚΟ
 - Φαγητά
 - Ορεκτικά
 - Τζατζίκι 3.50
 - Τυροκαφτερή 4.00
 - Σαλάτες
 - Πράσινη 5.20
 - Κόκκινη 6.00
 - Ανάμικτη 6.50
 - Κυρίως Πιάτα
 - Κρεατικά
 - Αρνάκι 8.50
 - Μοσχαραάκι 9.00
 - Γουρουνάκι 7.00
 - Λαδερά
 - Γεμιστά 5.00
 - Φασολάκια 4.50
 - Ζυμαρικά
 - Μακαρόνια 5.70
 - Κριθαράκι 6.00
 - Ψάρι Ημέρας 8.00
 - Πρασσόπιττα 4.00
 - Γλυκά
 - Παγωτά 3.50
 - Μπακλαβάς 2.50
 - Ποτά
 - Μη Αλκοολούχα Ποτά
 - Λεμονάδα 3.00
 - Πορτοκαλάδα 3.00
 - Κόκα Κόλα 3.00
 - Χυμοί 4.50
 - Μπίρες
 - Αμστελ 3.50
 - Μύθος 3.50
 - Corona 4.50
 - Breckenridge 1676 7.50
 - Κρασιά
 - Λευκά
 - Μοσχοφίλερο 8.50
 - Ασύρτικο 9.50
 - Ερυθρά
 - Ξινόμαυρο 9.90
 - Αγιωργήτικο 9.90
 - Ροζέ
 - Ερωδιός 8.50
 - Γλυκά
 - Σάμου 9.20
 - Λήμνου 9.00
 - Διάφορα Άλλα
 - Τσίπουρο 5.00
 - Ούισκι 6.00
 - Κονιάκ 6.00