

Εργαλεία Ανάπτυξης Λογισμικού και Προγραμματισμού Συστημάτων

Αναφορά 2ης άσκησης

Βασίλειος Λυμπεράκης 2015030183

Παναγιώτης Γιαννακόπουλος 2015030171

Ομάδα LAB21142559

Τμήματα προγράμματος

- Λειτουργία 1:
 - ο Υλοποίηση συνάρτησης εισαγωγής δεδομένων (Παναγιώτης Γιαννακόπουλος)
 - ο Υλοποίηση συναρτήσεων ελέγχου εγκυρότητας (Βασίλειος Λυμπεράκης)
 - ο Σχεδιασμός ροής ελέγχου απόδειξης (Παναγιώτης Γιαννακόπουλος & Βασίλειος Λυμπεράκης)
- Λειτουργία 2: (Παναγιώτης Γιαννακόπουλος)
- Λειτουργία 3: (Βασίλειος Λυμπεράκης)

Εισαγωγή αποδείξεων

Διαβάζουμε κάθε γραμμή από το αρχείο εκτελώντας του απαραίτητους ελέγχους έτσι ώστε να καταλάβουμε σε ποιο σημείο της απόδειξης βρισκόμαστε. Στην συνέχεια εκτελούνται οι απαραίτητοι έλεγχοι εγκυρότητας των δεδομένων που διαβάζουμε, όπως περιγράφονται παρακάτω. Σε περίπτωση επιτυχίας όλων των ελέγχων αποθηκεύουμε την απόδειξη και έπειτα επαναλαμβάνουμε την ίδια διαδικασία για την επόμενη απόδειξη.

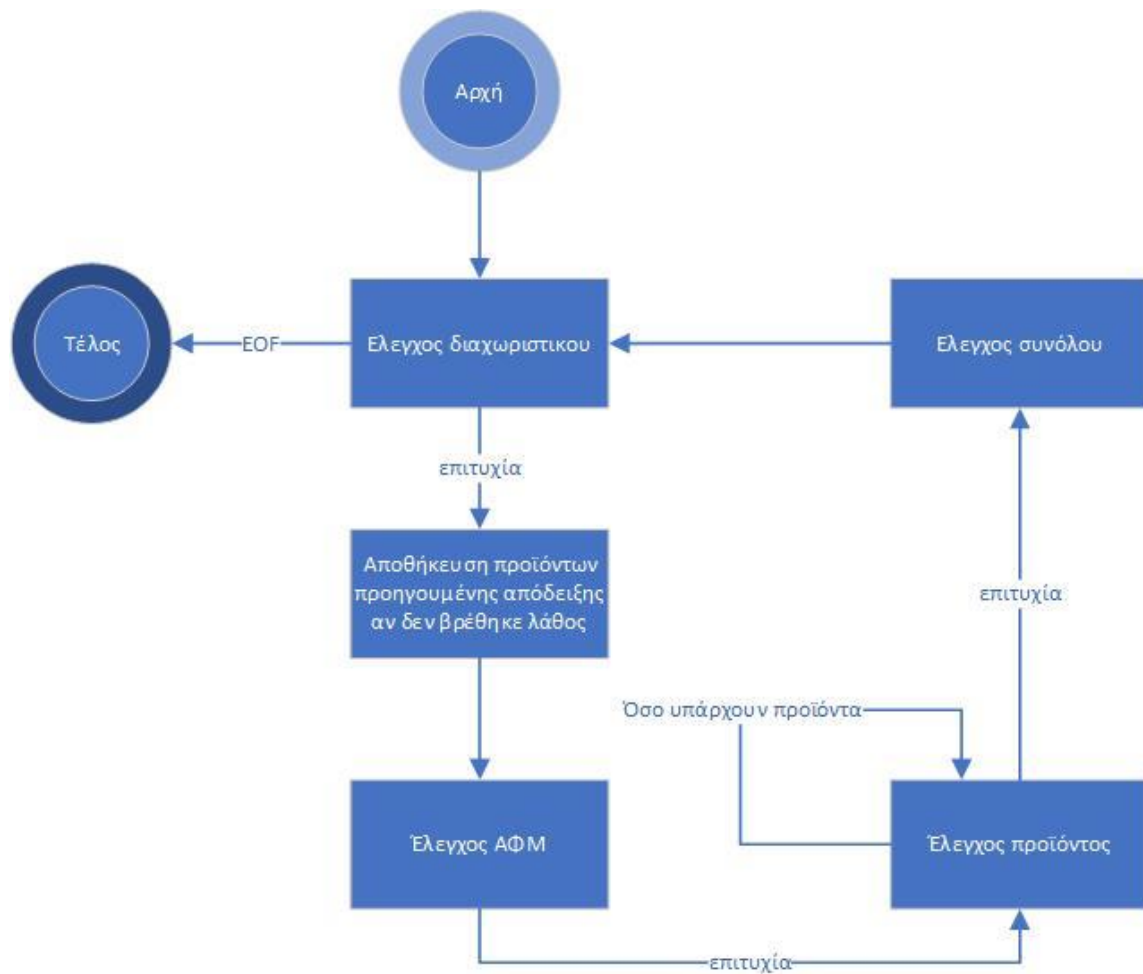
Έλεγχος απόδειξης

Για την σωστή αποθήκευση των δεδομένων πραγματοποιούνται διάφοροι έλεγχοι, ώστε να μην δημιουργείται κάποιο πρόβλημα στη μετέπειτα λειτουργία του. Οι έλεγχοι αυτοί είναι:

- Διαχωριστικού: Στον έλεγχο διαχωριστικού εξετάζεται ότι η γραμμή που διαχωρίζει τις αποδείξεις αποτελείται αυστηρά από χαρακτήρες '-'. Το μήκος της γραμμής μπορεί να είναι μεταβλητό, εφόσον δεν ορίζεται κάποιο σταθερό μέγεθος από την εκφώνηση. Σε περίπτωση λάθους απορρίπτονται δύο αποδείξεις, αυτή που προηγείται του διαχωριστικού και αυτή που βρίσκεται μετά.
- ΑΦΜ: Ο έλεγχος ΑΦΜ επιβεβαιώνει ότι υπάρχει το συγκεκριμένο πεδίο και ότι η ορθογραφία του είναι σωστή. Στη συνέχεια ελέγχει ότι η τιμή που δίνεται είναι δεκαψήφια και αποτελείται μόνο από αριθμούς. Αν κάτι από τα παραπάνω δεν τηρείται η απόδειξη αυτόματα απορρίπτεται.

- **Προϊόντος:** Στον έλεγχο προϊόντος εξετάζονται όλα τα πεδία που υπάρχουν σε μία γραμμή. Πιο συγκεκριμένα ελέγχεται ότι η ποσότητα είναι ένας θετικός ακέραιος αριθμός και τόσο η ανά μονάδα τιμή όσο και η συνολική τιμή είναι θετικοί δεκαδικοί αριθμοί, με μόνο δύο ψηφία στο δεκαδικό τους μέρος. Τέλος, ελέγχεται ότι το γινόμενο της ποσότητας με της ανά μονάδα τιμής ισούται με τη συνολική τιμή. Για την αποφυγή λαθών στη στρογγυλοποίηση υπάρχει μία ανοχή στη διαφοροποίηση των δύο τιμών μετά το τρίτο δεκαδικό ψηφίο.
- **Συνόλου:** Στον έλεγχο συνόλου επιβεβαιώνεται αρχικά ότι υπάρχει αυτό το πεδίο με τη σωστή ορθογραφία. Στη συνέχεια ελέγχεται ότι είναι δεκαδικός αριθμός με το δεκαδικό μέρος να είναι αυστηρά δύο ψηφίων. Τέλος, υπολογίζεται το άθροισμα των τιμών των προϊόντων και επιβεβαιώνεται ότι είναι ίδιο με το άθροισμα που εμφανίζεται.

Το διάγραμμα που ακολουθείται για τους παραπάνω ελέγχους φαίνεται στην παρακάτω εικόνα:



Δομή αποθήκευσης δεδομένων

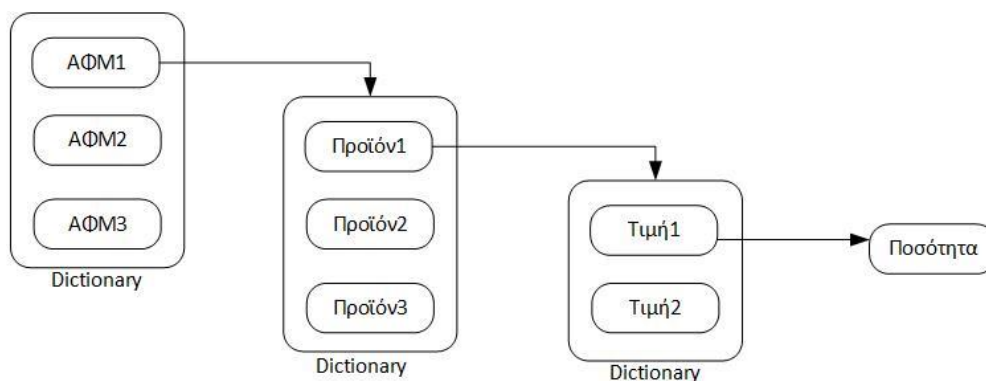
Αρχικά αναλύοντας τα δεδομένα μας συμπεραίνουμε ότι: ένα ΑΦΜ μπορεί να έχει ένα ή περισσότερα προϊόντα, ένα προϊόν μπορεί να έχει διαφορετικές τιμές, και για μια δεδομένη τιμή έχουμε μια συνολική ποσότητα που αγοράστηκε το προϊόν. Για την αποθήκευση των δεδομένων στην μνήμη του υπολογιστή χρησιμοποιήθηκε μια δομή με τρία εμφωλευμένα λεξικά.

Αναλυτικά, έχουμε ένα λεξικό το οποίο έχει ως κλειδιά τα ΑΦΜ που δείχνουν σε λεξικά από προϊόντα. Το λεξικό ενός προϊόντος έχει ως κλειδί το όνομα του προϊόντος και δείχνει σε ένα λεξικό για τις τιμές. Στην συνέχεια το λεξικό της τιμής κόστους δείχνει στην ποσότητα που αγοράστηκε με την συγκεκριμένη τιμή. Το συνολικό κόστος δεν αποθηκεύεται καθώς μπορεί να υπολογιστεί από το γινόμενο ποσότητας και τιμής.

Η επιλογή αυτή έγινε για τους εξής λόγους:

- Ικανοποιούνται τα παραπάνω χαρακτηριστικά των δεδομένων αποφεύγοντας περιττές επαναλήψεις ομοίων δεδομένων, για παράδειγμα αποθηκεύουμε το ΑΦΜ μόνο μια φορά και όχι όσες φορές είναι ο αριθμός των προϊόντων που του αντιστοιχούν.
- Τα λεξικά είναι πολύ ταχύτερα στην ανάκτηση, εισαγωγή και ανανέωση δεδομένων καθώς ισχύει $\text{dictionary}[\text{key}] = \text{value}$. Συγκεκριμένα έχουμε πολυπλοκότητα $O(1)$ για ανάκτηση/αναζήτηση δεδομένων ενώ σε έναν πίνακα θα είχαμε πολυπλοκότητα $O(N)$ όπου N το μέγεθος του πίνακα καθώς απαιτούνται βρόχοι. Συμπερασματικά με τα λεξικά επιτυγχάνουμε μεγαλύτερη ταχύτητα και πιο συνοπτικό κώδικα.

Παρακάτω φαίνεται η δομή σχηματικά:



Υποστήριξη πολύ μεγάλων αρχείων

Το πρόγραμμα μας μπορεί να εκτελέσει όλους του απαραίτητους ελέγχους και υπολογισμούς ακόμα και για πολύ μεγάλα αρχεία καθώς διαβάζουμε από το αρχείο μια γραμμή την φορά και μεταβαίνουμε στην εκάστοτε κατάσταση. Με αυτόν τον τρόπο για την εκτέλεση των ελέγχων πάνω στο αρχείο, δεσμεύεται ένα μικρό μέρος μνήμης, δηλαδή αυτό της μιας γραμμής.

Όσον αφορά την αποθήκευση των αποδείξεων στη δομή που αναλύθηκε παραπάνω, αποφεύγουμε σε ένα μεγάλο βαθμό την αποθήκευση διπλότυπων δεδομένων και δεδομένων που μπορούμε ευκολά να υπολογίσουμε ξανά (όπως το τελικό κόστος ενός ΑΦΜ για κάποιο προϊόν το οποίο υπολογίζεται από το γινόμενο του ποσοστού και της τιμής). Με αυτόν τον τρόπο επιτυγχάνουμε καλύτερη διαχείριση της μνήμης και αυξάνουμε τον όγκο διαφορετικών καταχωρήσεων που μπορούν να αποθηκευτούν.

Εμφάνιση στατιστικών

Ακολουθούν screenshots με την εκτέλεση του προγράμματος με την είσοδο που αναρτήθηκε στο courses

Λειτουργία 1:

Εισαγωγή αποδείξεων μέσω αρχείων στο σύστημα.

```
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 1
Enter the file name: veryEasyInput1
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 1
Enter the file name: veryEasyInput2
```

Λειτουργία 2:

Στη λειτουργία αυτή ζητείται από τον χρήστη να δώσει το όνομα ενός προϊόντος. Στη συνέχεια γίνεται μία αναζήτηση και εμφανίζονται όλα τα ΑΦΜ που έχουν πουλήσει αυτό το προϊόν μαζί με το άθροισμα των κερδών του καταστήματος.

```
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 2
Enter the product name: KPEATA
0123456789 100.00
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 2
Enter the product name: κρεατα
0123456789 100.00
```

```

MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 2
Enter the product name: ΧΩΡΙΑΤΙΚΗ
0123456788 22.00
0123456789 44.32
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 2
Enter the product name: ΑΠΑΚΙ
0123456789 16.02
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 2
Enter the product name: ΤΖΑΤΖΙΚΙ

```

Λειτουργία 3:

Στη λειτουργία αυτή ζητείται από τον χρήστη να δώσει ένα ΑΦΜ ενός μαγαζιού. Στη συνέχεια γίνεται μία αναζήτηση και εμφανίζονται όλα τα προϊόντα που έχει πουλήσει αυτό το μαγαζί μαζί με το άθροισμα των κερδών του καταστήματος. Η συνάρτηση που καλείται αρχικά ελέγχει την εγκυρότητα του ΑΦΜ μέσω του αντίστοιχου ελέγχου και σε περίπτωση επιβεβαίωσης συλλέγει τα δεδομένα ανά προϊόν και υπολογίζει τα συνολικά έσοδα.

```

MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 3
Enter AFM: 0123456787
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 3
Enter AFM: 0123456788
ΚΡΑΣΙ 8.20
ΜΠΟΥΡΕΚΙ 14.20
ΧΩΡΙΑΤΙΚΗ 22.00
MENU
1: read new input file
2: print statistics for a specific product
3: print statistics for a specific AFM
4: exit the program
Give your preference: 3
Enter AFM: 0123456789
ΑΠΑΚΙ 16.02
ΚΡΑΣΙ 28.00
ΚΡΕΑΤΑ 100.00
ΧΩΡΙΑΤΙΚΗ 44.32

```