

Ονοματεπώνυμο: Νίκος Κουγιανός
AM: 1115201100060
Unix ID: sdi1100060

Τμήμα Πληροφορικής και Τηλεπικοινωνιών **Προγραμματισμός Συστήματος / Περίοδος 2016-2017 / 1η Εργασία**

Το πρόγραμμα υλοποιεί όλα τα ερωτήματα της εκφώνησης της εργασίας. Η μεταγλώττιση του γίνεται με την εντολή `make`, με την χρήση κατάλληλου `MakeFile`. Η κλήση της εφαρμογής γίνεται με τον τρόπο που περιγράφεται στην εκφώνηση:

```
./werhauz -o Operations -h1 Hashtable1NumOfEntries -h2 Hashtable2NumOfEntries -s BucketSize
```

χωρίς να γίνεται η χρήση του προαιρετικού configuration file.

Αποτελείται από τα εξής .cpp αρχεία:

ArrayMaxHeap.cpp
Bucket.cpp
Buckets.cpp
CDR.cpp
CountryCodesList.cpp
HashTable.cpp
List.cpp
ListOfCDR.cpp
main.cpp
Manager.cpp
MaxHeap.cpp
Parse.cpp

Και απο τα εξής .h αρχεία:

ArrayMaxHeap.h
Bucket.h
Buckets.h
CDR.h
CountryCodesList.h
HashTable.h
List.h
ListOfCDR.h
Manager.h
MaxHeap.h
Parse.h

Ξεκινώντας από το χαμηλότερο επίπεδο (CDR) και πηγαίνοντας προς τα πάνω, η σειρά που ακολουθείται όσον αφορά τις δομές-κλάσεις είναι η εξής:
CDR->ListOfCDR->Bucket->Buckets->HashTable->Manager

- Στην κλάση CDR κρατάω αυτά που ζητάει και η εκφώνηση + ένα τιμολόγιο (δισδιάστατο πίνανα 4x4) το οποίο χρησιμοποιείται για τον υπολογισμό κόστους του εκάστοτε CDR.

Τους τηλεφωνικούς αριθμούς τους κρατάω σαν *long long*, και απλά όταν διαβάζω μια εγγραφή απο *input* διαγράφω την παύλα. Για την ημερομηνία/ώρα χρησιμοποιήθηκε το *struct tm* της *<ctime>*.

- Το ListOfCDR είναι μια λίστα που κάθε κόμβος της αποτελείται από ένα CDR, και δείκτες σε *previous*, *next*. Κάθε CDR λίστα είναι μοναδική και περιέχει όλα τα CDR που σχετίζονται με έναν συνδρομητή.

- Το Bucket είναι ένας πίνακας δεικτών που ο καθένας δείχνει σε μια διαφορετική CDR λίστα.
- Το Buckets είναι μια λίστα που κάθε κόμβος της αποτελείται από Bucket (data) και δείκτες σε previous, next.
- Το HashTable κρατάει έναν πίνακα δεικτών που ο καθένας δείχνει σε ένα διαφορετικό Buckets.
- Τέλος ο Manager που μέσω κατάλληλων συναρτήσεων μπορεί να διαχειρίζεται και τα 2 HashTables (originators, destinations) καθώς και ό,τι αυτά περιλαμβάνουν. Διαχειρίζεται επίσης και την max Heap.

- Η max Heap έχει υλοποιηθεί με τον εξής τρόπο:

Σε κάθε CDR λίστα (που αντιστοιχεί και σε κάθε πελάτη) κρατάω και έναν double amount που είναι το ποσό που έχει ξοδέψει ο πελάτης εκείνη την στιγμή. Κάθε φορά που γίνεται insert ή delete ενός CDR το amount αυτό ενημερώνεται κατάλληλα. Οπότε η max Heap είναι μια λίστα που λειτουργεί σαν δέντρο (έστω είμαστε στον κόμβο i, ο κόμβος i+1 είναι το αριστερό παιδί και ο κόμβος i+2 είναι το δεξί παιδί).

Μέσω κατάλληλων συναρτήσεων γίνεται ταξινόμηση στους κόμβους με αποτέλεσμα ο πρώτος κόμβος να έχει πάντα το μεγαλύτερο amount κ.ο.κ.

Ο κώδικας για την max Heap είναι από την πηγή

<https://www.codeproject.com/Tips/816934/Min-Binary-Heap-Implementation-in-Cplusplus>
διαμορφωμένος κατάλληλα για να λειτουργεί στο δικό μου πρόβλημα.

- Το CountryCodesList είναι μια λίστα που περιέχει τους τριψήφιους κωδικούς των χωρών, και το List είναι μια γενική βοηθητική λίστα που κρατάει double και χρησιμοποιείται στην max Heap.

-Το Parse διαβάζει κατάλληλα τις εντολές από ένα input file, είτε από το command line.

Οι περισσότερες συναρτήσεις των υψηλών επιπέδων καλούν τις συναρτήσεις των κλάσεων που περιέχουν. Παράδειγμα η εντολή find, η οποία καλεί την συνάρτηση find του Manager, που καλεί την find του HashTable1, το οποίο καλεί την find του Bucket κ.ο.κ. Εν κληθεί μια συνάρτηση χαμηλού επιπέδου (επιπέδου CDR List) η οποία θα μας επιστρέψει την κατάλληλη λίστα.

Το πρόγραμμα δεν τερματίζει με την εντολή bye (αυτή απλά διαγράφει τα πάντα και δημιουργεί ξανα νέες δομές) αλλά με την εντολή end, η οποία exit από το πρόγραμμα.