

Πανεπιστήμιο Πατρών

Τμήμα Μηχανικών ΗΥ κ Πληροφορικής

---

## **Δομές Δεδομένων Εργασία 2014-2015**

---

*Ιωάννα Τζανέτου*  
tzanetou  
5084

June, 2015



---

## Εισαγωγή

Το πρότζεκτ υλοποιήθηκε με Python 2.7 και δοκιμάστηκε σε περιβάλλον Windows 7. Για την παραγωγή των γραφημάτων του πέμπτου ερωτήματος χρησιμοποιήθηκε το matplotlib. Αν εξαιρεθεί το plotting οποιοσδήποτε python interpreter αρκεί.

## A Ερώτημα

Στο ερώτημα αυτό ζητείται να εκτυπωθεί ένα μενού από το οποίο ο χρήστης επιλέγει τη λειτουργία που θέλει να εκτέλεσει πάνω στο αρχείο data.csv το οποίο δίνεται από τη σελίδα του μαθήματος.

Προϋπόθεση είναι αρχικά να γίνουν load οι εταιρίες μέσω της επιλογής 1. Στο μενού έχει προστεθεί μία επιπλέον δυνατότητα για φόρτωση ενός μικρότερου κομματιού του αρχείου ώστε η εκτέλεση του προγράμματος να γίνεται σε μικρότερο χρόνο.

Τα παραπάνω υλοποιούνται στο module main.py το οποίο χρησιμοποιεί το companyfuncs.py. Η γραμμική αναζήτηση επιτυγχάνεται με βρόχους for πάνω στη λίστα.

## B Ερώτημα

Στο ερώτημα αυτό εφαρμόζουμε δυαδική αναζήτηση που υλοποιείται στη συνάρτηση binary\_search μέσα στην κλάση Companies. Για την αναζήτηση πρέπει να διατηρείται η λίστα εταιριών ταξινομημένη μέσω της λίστας argsorted. Σε περίπτωση διαγραφής/εισαγωγής γίνεται ξανά ταξινόμηση της λίστας.

## Γ Ερώτημα

Υλοποιούμε ένα κομβοπροσανατολισμένο AVL δέντρο στο module AVL.py με βάση τα id των εταιριών. Σε κάθε πράξη ελέγχεται η ζύγιση του υποδέντρου και αν έχει χαλάσει γίνονται οι κατάλληλες επαναζυγιστικές κινήσεις.

## Δ Ερώτημα

Τα tries βρίσκονται στο module trie.py. Η οργάνωση της βάσης γίνεται βάσει επιθέτου. Αν υπάρχει συνωνυμία επιστρέφονται όλες οι εταιρίες που απασχολούν το όνομα αυτό.

---

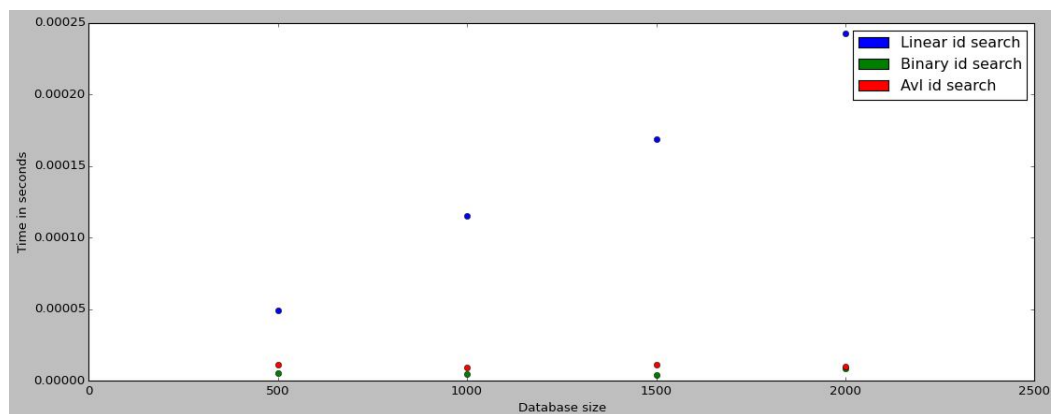
## Ε Ερώτημα

Στο ερώτημα αυτό εξετάζουμε την απόδοση κάθε μεθόδου. Για να είναι αξιόπιστες οι παρατηρήσεις, για κάθε μέθοδο χρονομετρούμε 10 φορές κάθε πείραμα συγκεκριμένου μεγέθους και παίρνουμε το μέσο όρο.

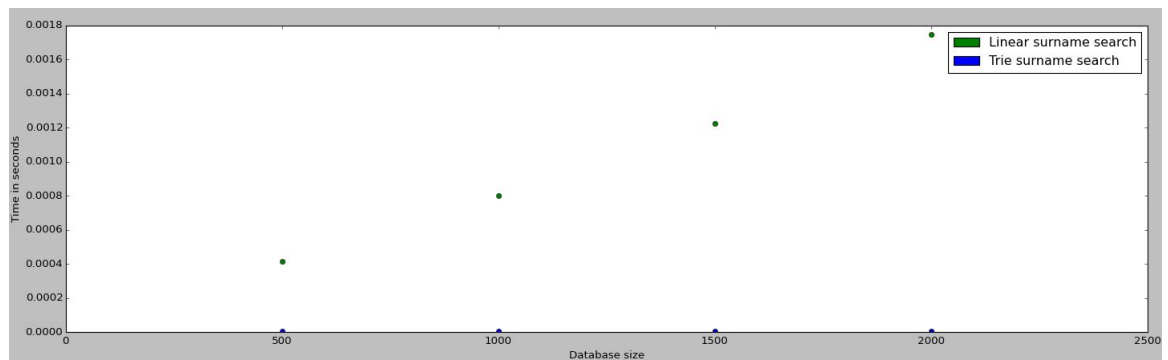
Το αρχείο `search.py` περιέχει τη συνάρτηση `ioannaki` που δημιουργεί μια λίστα (ψευδοτυχαίων) εταιριών προς αναζήτηση. Αφού δημιουργηθεί αυτή η λίστα το `search.py` αναλαμβάνει να αναζητήσει την παραπάνω λίστα στη βάση που διαθέτουμε. Το μέγεθος της βάσης το μεταβάλλουμε για να δούμε την απόδοση των αλγορίθμων ανάλογα με το μέγεθος και για κάθε μέγεθος βάσης παίρνουμε το μέσο χρόνο αναζήτησης.

Όπως αναφέρθηκε παραπάνω για την οπτικοποίηση των αποτελεσμάτων χρησιμοποιήσαμε το module `matplotlib` ώστε να εξάγουμε ευκολότερα συμπεράσματα για τις διαφορές μεταξύ των μεθόδων.

Η εκτέλεση του `search.py` δίνοντας ως παράμετρο `-p` αν υπάρχει εγκατεστημένη στο σύστημα η `matplotlib` ώστε να δημιουργηθούν τα παράθυρα γραφημάτων.



Από το παραπάνω γράφημα παρατηρούμε ότι η γραμμική αναζήτηση αυξάνει γραμμικά με την αύξηση του μεγέθους της βάσης. Βλέπουμε όμως ότι δεν πρόκειται για ευθεία γραμμή καθώς υπάρχουν κάποιες αυξομειώσεις γιατί μερικές αναζητήσεις μπορεί να είναι πιο "τυχερές". Αντίθετα η δυαδική αναζήτηση και η αναζήτηση με `avl` κινούνται σε λογαριθμικούς χρόνους όπως φαίνεται στο σχήμα και επιβεβαιώνεται με τη θεωρία και η αύξηση σε χρόνο αναζήτησης όσο μεγαλώνει η βάση είναι μικρή.



Στην αναζήτηση με βάση το επώνυμο παρατηρούμε διαφορές καθώς τώρα πρέπει να προσπελαύνουμε όλη τη λίστα ακόμη και αν η αναζήτησή μας είναι επιτυχής στο πρώτο βήμα, διότι τώρα το επώνυμο δεν αποτελεί μοναδικό κλειδί.

Λόγω της επιπλέον πολυπλοκότητας που προστέθηκε, η γραφική παράσταση δεν έχει πάλι τελείως γραμμική μορφή και η διαφορά τώρα στους χρόνους είναι πολύ μεγαλύτερη.

Το trie δεν επηρεάζεται από αυτό το πρόβλημα καθώς κατά την κατασκευή του μέσω της συνάρτησης `build_employees` που έχει αποθηκευτεί σε κάθε κλειδί μια λίστα με τις εταιρίες στις οποίες ανήκει αυτό το επίθετο.

Για την αρχικοποίηση λοιπόν του trie πρέπει να περάσουμε μια φορά τη λίστα αλλά τελικά ο χρόνος για να φτάσουμε σε ένα κλειδί δεν εξαρτάται από το μέγεθος της βάσης αλλά από το μήκος του κλειδιού. Συνεπώς ο μέσος χρόνος αναζήτησης είναι σχεδόν σταθερός.

## Παρατηρήσεις

Γενικά παρατηρούμε ότι η απόδοση των γραμμικών αναζητήσεων πέφτει όσο μεγαλώνει το μέγεθος της βάσης σε αντίθεση με τις αναζητήσεις που έχουν βάση τα AVL ή τα tries και τη δυαδική αναζήτηση.

Επιπλέον τα γραφήματα δεν επαληθεύουν τη θεωρητική τιμή. Αυτό μπορεί να οφείλεται στο ότι τα δείγματα είναι ψευδοτυχαία και στις παραπάνω λειτουργίες που προσθέσαμε για μεγαλύτερη ευκολία στη χρήση της βάσης.

Παρόλα αυτά είναι βασικό να συγκρατήσουμε ότι τα tries συγκεκριμένα διαφοροποιούνται σημαντικά με τη γραμμική αναζήτηση και οι χρόνοι τείνουν στο μηδέν.

## 1 Βιβλιογραφία

<https://en.wikipedia.org/wiki/Trie>

<https://github.com/pgrafov/python-avl-tree/blob/master/pyavltree.py>