

3^η Άσκηση ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ 2010-11

Οργάνωση 10.000 Ονομάτων με ΑΤΔ BulkTree (χρήση ΔΔΑ)

Σας δίνονται 2 αρχεία με 10.000 άτομα το καθένα. Τα άτομα προσδιορίζονται με «επίθετο όνομα». Τα άτομα είναι σε τυχαία και αλφαβητική σειρά αντίστοιχα (αρχεία atoma-random.txt και atoma-sorted.txt). Επίσης δίνεται το αρχείο atoma-search.txt που περιέχει άτομα προς αναζήτηση.

Κατασκευάστε το ΑΤΔ BulkTree με σκοπό την μαζική διαχείριση ατόμων με ΔΔΑ. Η δομή είναι ένα ΔΔΑ σε κάθε κόμβο του οποίου φυλάσσεται ένα άτομο. Ακολουθούν οι πράξεις του BulkTree

1. **Δημιουργία.** Δημιουργεί ένα κενό BulkTree.
2. **Κατασκευή από αρχείο.** Διαβάζει άτομα από αρχείο και τα εισάγει στο BulkTree. Επίσης επιστρέφει σε πίνακα τους χρόνους εισαγωγής μετά από 2, 4, 8, 16, ... , 8192 άτομα και μετά από όλα τα άτομα.
3. **Εκτύπωση σε αρχείο.** Διατρέχει το BulkTree και εκτυπώνει τα άτομα με αλφαβητική σειρά σε αρχείο. Επίσης επιστρέφει τον συνολικό χρόνο που απαιτήθηκε.
4. **Αναζήτηση από αρχείο.** Διαβάζει άτομα προς αναζήτηση από ένα αρχείο και τα αναζητεί διαδοχικά στο BulkTree. Επιστρέφει το πλήθος των ατόμων που βρέθηκαν και τον συνολικό χρόνο που απαιτήθηκε.
5. **Καταστροφή.** Καταστρέφει όλους τους κόμβους του BulkTree.

Οι παραπάνω πράξεις πρέπει να υλοποιηθούν 2 φορές χρησιμοποιώντας πράξεις των ΑΤΔ Απλό ΔΔΑ και AVL ΔΔΑ αντίστοιχα, οι υλοποιήσεις των οποίων σας δίνονται στα αντίστοιχα ζεύγη αρχείων (.c ,.h). Ως οδηγό της τεχνικής μπορείτε να έχετε την υλοποίηση στοίβας με χρήση του ΑΤΔ Λίστα, 7.8.3 των πρότυπων υλοποιήσεων. Οι δυο υλοποιήσεις έχουν παρόμοια διεπαφή. Οι διαφορές είναι α) ότι οι πράξεις του απλού ΔΔΑ έχουν το πρόθεμα Tree ενώ του AVL ΔΔΑ το AVLTree και β) οι πράξεις εισαγωγής και διαγραφής σε AVL χρειάζονται μια *int παράμετρο επιπλέον.

Ζητείται επίσης να υλοποιήσετε το κυρίως πρόγραμμα, το οποίο

1. να δημιουργεί ένα BulkTree,
2. να κατασκευάζει ένα BulkTree από το αρχείο ονομάτων atoma-random.txt και να εκτυπώνει τους χρόνους μερικής και πλήρους κατασκευής,
3. να εκτυπώνει αλφαβητικά τα ονόματα σε αρχείο atoma-random-out.txt και τον χρόνο που απαιτήθηκε.
4. να αναζητά στο BulkTree τα άτομα από το αρχείο atoma-search.txt και να εκτυπώνει τον συνολικό χρόνο αναζήτησης καθώς και το πλήθος των ατόμων που βρέθηκαν.
5. να καταστρέφει πλήρως το BulkTree.
6. Να επαναλάβετε τα ζητούμενα 1-5 διαβάζοντας όμως ονόματα από το atoma-sorted.txt.

Να εκτελέσετε το κυρίως πρόγραμμα με τις δυο υλοποιήσεις του BulkTree (απλό ΔΔΑ και AVL ΔΔΑ).

Προσοχή, δεν πρέπει να κάνετε καμία αλλαγή στα αρχεία υλοποίησης απλό ΔΔΑ και AVL ΔΔΑ. Απαιτείται επίσης να αναπτύξετε νέο τύπο στοιχείου με πρότυπο τον τύπο στοιχείου που σας έχει δοθεί.

Οδηγίες Σχεδίασης και Ανάπτυξης Προγράμματος

Το πρόγραμμά σας πρέπει να είναι οργανωμένο σε ενότητες (modules) και σε πρόγραμμα-πελάτη. Εσείς πρέπει να αναπτύξετε το BulkTree.h, BulkTreeSimple.c, BulkTreeAVL.c (οι δύο εναλλακτικές υλοποιήσεις του BulkTree.h), main.c, TyposStoixeiou.h, TyposStoixeiou.c και να χρησιμοποιήσετε τις υλοποιήσεις των απλό ΔΔΑ και AVL ΔΔΑ .

Στη ζητείται επίσης να συμπεριλάβετε στην τεκμηρίωση συμπληρωμένο τον ακόλουθο πίνακα χρονομετρήσεων σας καθώς και σχολιασμό των χρόνων σε σχέση με τους θεωρητικούς χρόνους.

Εργασία/Χρόνος	Απλό ΔΔΑ- random input	Απλό ΔΔΑ- sorted input	AVL ΔΔΑ- random input	AVL ΔΔΑ- sorted input
Κατασκευή μετά από				
2				
4				
+++++	+++++	+++++	+++++	+++++
8196				
όλα				
Εκτύπωση				
Αναζήτηση				

Για την χρονομέτρηση δίνονται στον κατάλογο TimingDemos 3 προγράμματα επίδειξης: το SecResTiming.c χρησιμοποιεί την βιβλιοθήκη της C με ακρίβεια δευτερολέπτου και τα άλλα δύο WmSecResTiming.c και LmSecResTiming.c βιβλιοθήκες των Windows και Linux αντίστοιχα για ακρίβεια msec, που μάλλον θα σας χρειαστεί.

Παραδοτέα και Οδηγίες Παράδοσης

1. Πηγαίος κώδικας, δηλαδή BulkTree.h, BulkTreeSimple.c, BulkTreeAVL.c, main.c, TyposStoixeiou.h, TyposStoixeiou.c (Τα main των δοκιμών και οι δοθείσες υλοποιήσεις των ΔΔΑ δεν παραδίδονται)
2. Τεκμηρίωση (μέγιστο 1 σελίδα). Σύντομο κείμενο (txt ή pdf) με την εξής δομή:
 - Τα στοιχεία σας: (Όνομα-Επώνυμο-ΑΜ)
 - Λειτουργικότητα: Να περιγράψετε τι κάνει 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 Dev C++ σε περιβάλλον Windows XP, δοκιμάστηκε επίσης σε gcc και Unix.
 - Ο πίνακας χρόνων και ο σχολιασμός σας

Το αρχείο τεκμηρίωσης μαζί με τα πηγαία αρχεία του προγράμματος να τα βάλετε σε έναν φάκελο (directory), τον οποίο θα συμπιέσετε (zip, rar) και θα ανεβάσετε στο eclass. (Όσοι αναπτύσσετε σε dev να συμπεριλάβετε και το .dev). Προσοχή ανεβάστε το στην κατάλληλη κατηγορία υλοποίησης (Dev-C++ ή gcc). **Ημερομηνία Παράδοσης Κυριακή 29/5/2011.**

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 μηδενιστούν και δεν θα ληφθεί υπόψη ο βαθμός των γραπτών. Θα αξιολογηθούν η λειτουργικότητα, η δομή και η τεκμηρίωση του προγράμματος. Αναλυτικά:

Λειτουργικότητα (70/100)

- | | |
|----------------------------------|---------|
| 1. BulkTree Δημιουργία | (05/80) |
| 2. BulkTree Κατασκευή από αρχείο | (15/80) |
| 3. BulkTree Εκτύπωση σε αρχείο | (10/80) |
| 4. BulkTree Αναζήτηση από αρχείο | (15/80) |
| 5. BulkTree Καταστροφή | (05/80) |
| 6. Κυρίως Πρόγραμμα | (10/80) |
| 7. Τύπος στοιχείου | (10/80) |

Δομή (15/100)

- | | |
|---|---------|
| 8. Υλοποίηση με απλό ΔΔΑ και AVL ΔΔΑ | (10/15) |
| 9. Δομημένο Πρόγραμμα-πελάτη (μορφοποίηση, σχόλια, κλπ) | (05/15) |

Τεκμηρίωση και Παρουσίαση (15/100)

- | | |
|--------------------------|---------|
| 10. Πίνακας Χρόνων | (05/15) |
| 11. Σχολιασμός | (05/15) |
| 12. Υπόλοιπο Τεκμηρίωσης | (05/15) |

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.