



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΗΜΜΥ

ΜΑΘΗΜΑ: ΠΛΗ 201 – ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ
ΑΡΧΕΙΩΝ

ΕΡΓΑΣΤΗΡΙΟ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΡΧΕΙΩΝ

Αναφορά 1^{ου} Εργαστηρίου :

Όνομα: Παπουτσάκης Νικόλαος 2019030206

Εισαγωγή :

Στο 1^ο εργαστήριο ήρθαμε σε επαφή με την διαχείριση αρχείων στο δίσκο, αλλά και στην κεντρική μνήμη και έγινε επεξεργασία δεδομένων που περιείχαν αυτά τα αρχεία. Για την εύκολη κατανόηση του τί γίνεται και του τί θέμε να κάνουμε σε αυτήν την άσκηση, θα χωρίσουμε το συγκεκριμένο project σε 2 μέρη. Πρώτον, την υλοποίηση σημαντικών «εργαλείων» που θα βοηθήσουν στην επεξεργασία των αρχείων και δεύτερον το Search Testing, με το οποίο θα πραγματοποιήσουμε κάποια search tests για να συλλέξουμε κάποια δεδομένα που θέλουμε.

1^η Φάση : Υλοποίηση

Αρχικά κληθήκαμε να κατασκευάσουμε μία κλάση **File Manager** η οποία θα ήταν εκείνη που θα διαχειριζόταν ένα αρχείο. Υλοποιήθηκαν σημαντικές μέθοδοι (Read Block, Write Block, Append Block e.t.c) που μας φάνηκαν χρήσιμες στο υπόλοιπο της άσκησης.

Έπειτα θέλαμε να δημιουργήσουμε αρχεία με 4 διαφορετικούς τρόπους οργάνωσης της πληροφορίας τους. Οι κλάσεις Serialized File Creator και Sorted File Creator, κάνουν ακριβώς αυτή τη δουλειά.

- Η **Serialized File Creator** δημιουργεί ένα νέο αρχείο και αποθηκεύει σε αυτό 10.000 Nodes με τυχαία σειρά, αλλά και τυχαία κλειδιά σε κάθε node. Είναι ιδιαίτερα σημαντική στον Α τρόπο, αλλά και στον Β αφού περιέχει ένα Index Node[] στο οποίο αποθηκεύονται nodes που περιέχουν τόσο το κλειδί όσο και τη σελίδα που βρίσκεται ένα Data Node. Η διαδικασία αποθήκευσης του κλειδιού από κάθε Node αλλά και εύρεσης της σελίδας της οποίας βρίσκεται το συγκεκριμένο Node, γίνεται ταυτόχρονα κατά την δημιουργία του 1^{ου} αρχείου (Α τρόπος οργάνωσης).

- Η **Sorted File Creator** δημιουργεί ένα νέο αρχείο από το ήδη υπάρχον αρχείο του Α και Β τρόπου οργάνωσης και αποθηκεύει σε αυτό 10.000 Nodes τα οποία είναι ταξινομημένα με βάση το κλειδί.

2^η Φάση : Search Testing

Αφού δημιουργήσουμε τα παραπάνω αρχεία, έπειτα υλοποιήσαμε τις μεθόδους αναζήτησης Array Search και Binary Search. Δύο ξεχωριστές κλάσεις όπου η κάθε μια περιέχει μια μέθοδο αναζήτησης (search) για κάθε τρόπο οργάνωσης της πληροφορίας του αρχείου.

Τέλος με την χρήση των κλάσεων Array Search Tester και Binary Search Tester πραγματοποιήθηκαν τα search tests για την μέτρηση της απόδοσης κάθε τρόπου οργάνωσης του αρχείου σε αριθμό προσβάσεων δίσκου. Κάθε κλάση περιέχει μία μέθοδο do Test() η οποία ξεκινάει το test, κάνει 20 αναζητήσεις με τυχαίο κλειδί κάθε φορά και υπολογίζει τον μέσο όρο αριθμού προσβάσεων δίσκου.

Τεκμηρίωση των Αποτελεσμάτων :

Μέθοδος	Α. τρόπος οργάνωσης αρχείου	Β. τρόπος οργάνωσης αρχείου	Γ. τρόπος οργάνωσης αρχείου	Δ. τρόπος οργάνωσης αρχείου	Απόδοση Εξωτερικής Ταξινόμησης περίπτωση Γ	Απόδοση Εξωτερικής Ταξινόμησης περίπτωση Δ
Απόδοση (αριθμός προσβάσεων δίσκου)	1139	291	10	8	5000	1250

Σχόλια:

Η μέθοδος Β είναι προφανώς πιο αποδοτική σε σχέση με την Α, αφού το 2^ο αρχείο έχει λιγότερα pages. Άρα κατά μέσο όρο, η σειριακή αναζήτηση στην Β μέθοδο διασχίζει λιγότερα pages.

Για τις Γ και Δ μεθόδους προφανώς, η δυαδική αναζήτηση είναι πολύ πιο γρήγορη σε σχέση με την σειριακή. Παρατηρούμε πως κατά μέσο όρο οι προσβάσεις στο δίσκο έχουν μικρή διαφορά. Αυτό συμβαίνει διότι πάλι το αρχείο των δεικτών είναι μικρότερο σε μέγεθος από το αρχείο με τα Nodes.

Η απόδοση του external sorting δεν βγάζει νόημα στην περίπτωση μας, αφού ολόκληρο το αρχείο (αρχείο κλειδιών και Data Nodes) διαβάζεται στον buffer της κεντρικής μνήμης και έπειτα sort-αρεται. Άρα τα disk accesses θα είναι για κάθε αρχείο 2 φορές τον αριθμό των σελίδων του, μία φορά για read και μία φορά για write.

ΤΕΛΟΣ ΕΡΓΑΣΤΗΡΙΑΚΗΣ ΑΝΑΦΟΡΑΣ