Δομές δεδομένων και αρχείων

Αναφορά 1ης εργασίας

Θωμάς Λάγκαλης

Μάρτιος 2023

1 Διαγράμματα

Στα διαγράμματα γίνεται αντιληπτό πως η μορφή των καμπυλών για τους τρόπους Α και Β είναι γραμμική ενώ για τον τρόπο Γ λογαριθμική. Το διάγραμμα της μεθόδου Β περιέχει γραμμική καμπύλη μικρότερης κλίση από αυτήν της μεθόδου Α. Αυτό συμβαίνει διότι η σελίδα του αρχείο με τα ζεύγη κλειδιών-δεικτών σελίδας περιέχει περισσότερα στιγμιότυπα από το αρχείο με τα στιγμιότυπα DataClass . Έτσι, η αναζήτηση στο αρχείο κλειδιών-δεικτών του τρόπου Β εμπεριέχει λιγότερες προσβάσεις στον δίσκο από την μέθοδο Α που η αναζήτηση γίνεται σε αρχείο με τα στιγμιότυπα DataClass.

Στους τρόπους Α και Β η αναζήτηση γίνεται σειριακά, δηλαδή αν υπάρχουν Ν σελίδες στον δίσκο στη χειρότερη περίπτωση θα υπάρξουν Ν προσβάσεις στον δίσκο (N+1 για τον Β τρόπο γιατί θα γίνει μία επιπλέον πρόσβαση στο αρχείο με τα στιγμιότυπα DataClass όταν βρεθεί το κλειδί). Με την παραπάνω λογική προκύπτει ότι η πολυπλοκότητα για τους τρόπους Α και Β είναι O(N). Όσο αυξάνεται το πλήθος των εγγραφών, οι προσβάσεις στον δίσκο αυξάνονται γραμμικά. Στον Γ τρόπο υλοποιείται ο αλγόριθμος Binary Search σε ένα ταξινομημένο αρχείο κλειδιών-δεικτών ως προς τα κλειδιά. Ο αλγόριθμος αυτός όπως περιγράφεται και στην εκφώνηση υποδιπλασιάζει σε κάθε επανάληψή του το εύρος στο οποία κάνει την αναζήτηση. Εφαρμόζοντας ασυμπτωτική ανάλυση προκύπτει πως η πολυπλοκότητά του είναι O(logN), πράγμα που σημαίνει πως όσο αυξάνεται το πλήθος εγγραφών, οι προσβάσεις αυξάνονται λογαριθμικά. Επομένως, όπως φαίνεται και από τα διαγράμματα η αναζήτηση του τρόπου Γ είναι η πιο αποτελεσματική από άποψη αριθμού προσβάσεων και χρόνου εκτέλεσης, έπειτα είναι ο τρόπος Β και πιο αναποτελεσματικός είναι ο τρόπος Α.

Να τονιστεί πως ο χρόνος εκτέλεσης είναι ανάλογος του αριθμού προσβάσεων στη μνήμη. Δηλαδή, εφόσον ο χρόνος για την πρόσβαση στο δίσκο είναι Τ (σταθερός) τότε ο χρόνος εκτέλεσης του εκάσθωτε αλγορίθμου θα είναι $K \cdot T$ όπου Κ ο αριθμός προσβάσεων. Συνεπώς, τα η μορφή (γραμμική, λογαριθμική) των διαγραμμάτων για τον χρόνο εκτέλεσης-αριθμός εγγραφών είναι ίδια με τα διαγράμματα: αριθμός προσβάσεων-αριθμός εγγραφών.

Σημείωση: Οι εικόνες των διαγραμμάτων για να είναι πιο ευκρινείς βρίσκονται στο αρχείο με όνομα 31bytesAll.png και 59bytesAll.png για τις περιπτώσεις που το κάθε στιγμιότυπο είναι μεγέθους 31 bytes και 59 bytes αντίστοιχα. Επίσης, μπορούν να βρεθούν και στο αρχείο excel μαζί με το output που παρήγαγε το πρόγραμμα και στο οποίο βασίστηκαν.

2 Περιγραφή Κώδικα

Ο κώδικας αποτελείται από τις κλάσεις: **DataClass, DataPage, KeyPage, και Main**.

2.1 DataClass

Η κλάση αυτή αναπαριστά τα δεδομένα σύμφωνα με της εκφώνηση. Συγκεκριμένα, περιέχει δύο πεδία: έναν ακέραιο, το μοναδικό κλειδί και ένα πεδίο String που αντιστοιχεί στα δεδομένα (τυχαίο αλφαριθμητικό). Τα στιγμιότυπα της κλάσης αυτής αρχικοποιούνται με constructor που περιέχει ως όρισμα το κλειδί (ακέραιος) και το αλφαριθμητικό (String). Το μέγεθος του αλφαριθμητικού καθορίζεται με τρόπο που θα αναλυθεί στη συνέχεια. Η κλάση αποτελείται από τις εξής συναρτήσεις (μεθόδους):

- Getters/Setters
- toByteArray(int chunkSize): Λαμβάνει ως όρισμα το μέγεθος σε bytes των δεδομένων, ανάλογα με την περίπτωση της εκφώνησης (59 και 31 bytes) και επιστρέφει ένα array από bytes των δεδομένων του συγκεκριμένου στιγμιότυπου. Αυτό επιτυγχάνεται με την κλάση **java.nio.ByteBuffer**

2.2 DataPage

Η κλάση DataPage αναπαριστά την σελίδα στον δίσκο. Έχει δύο πεδία: ένα array από bytes (data) με τα δεδομένα και έναν ακέραιο αριθμό (dataSize) που δηλώνει το μέγεθος της σελίδας. Ο constructor λαμβάνει όρισμα το byte array με τα δεδομένα και αρχικοποιεί το dataSize με τιμή το μέγεθος του array που δόθηκε ως όρισμα. Οι συναρτήσεις που αποτελούν την κλάση είναι μόνο οι getters και οι setters.

2.3 KeyPage

Αναπαριστά τα ζεύγη κλειδιών-αριθμός σελίδας. Η κλάση αυτή χρησιμοποιείται στους τρόπους Β και Γ. Περιέχει τα πεδία: ένας ακέραιος για το κλειδί (key) και ένας για τον αριθμό της σελίδας στην οποία βρίσκεται η εγγραφή με το συγκεκριμένο κλειδί(pageIndex). Ο constructor λαμβάνει όρισμα τους δύο ακεραίους για το κλειδί και τον αριθμό σελίδας τους οποίους και αρχικοποιεί. Η κλάση εμπεριέχει επίσης και την μέθοδο toByteArray() η οποία είναι ίδια με την αντίστοιχη μέθοδο της DataClass με διαφορά ότι δεν έχει όρισμα διότι το μέγεθος των δεδομένων την κλάσης KeyPage είναι σταθερό και ίσο με 8 bytes (δύο ακέραιοι αριθμοί). Τέλος, η κλάση υλοποιεί το interface Comperable ώστε οι συγκρίσεις για την αναζήτηση με Binary Search (Γ τρόπος) να γίνονται με βάση το κλειδί, για αύτο και η μέθοδος compareTo().

2.4 Main

Στη κλάση Main υλοποιήται το μεγαλύτερο μέρος της λειτουργικότητας της εργασίας. Παρακάτω περιγράφονται οι μέθοδοί της.

- **pageToDataClass()**: Λαμβάνει όρισμα ένα DataPage στιγμιότυπο και τον μέγεθος σε bytes κάθε εγγραφής και επιστρέφει ένα array από στιγμιότυπα DataClass με τα δεδομένα της σελίδας.
- **pageToKeyPage()**: Κάνει την ίδια δουλειά με την pageToDataClass όμως επιστρέφει array με KeyPage στιγμιότυπα.
- **toData()**: Παίρνει παράμετρο ένα byte array (όπως διαβάζεται από τον δίσκο) το οποίο περιέχει τα δεδομένα μίας εγγραφής και επιστρέφει το αντίστοιχο στιγμιότυπο DataClass.
- **toKeyPagePair()**: Κάνει την ίδια δουλειά με την toData() όμως μετατρέπει τα δεδομένα σε KeyPage στιγμιότυπο.
- **readFromDisk()**: Διαβάζει από τον δίδκο μία σελίδα (σύμφωνα με τα πρότυπα του προγράμματος). Παίρνει όρισμα το String με το όνομα του αρχείου και την θέση από όπου θα διαβάσει (ακέραιος) και επιστρέφει ένα byte array με τα δεδομένα που διαβάστηκαν.
- **writeToDisk()**: Γράφει στον δίσκο ένα πλήθος από σελίδες. Παίρνει ορίσματα το όνομα του αρχείο στο οποίο θα γίνει η εγγραφή (String) και ένα array από DataPage στιγμιότυπα και επιστρέφει το μέγεθος του DataPage array αν η εγγραφή γίνει επιτυχώς ή -1 αν υπάρξει πρόβλημα (exception).

- **toDiskPageRecord()**: Μετατρέπει ένα byte array σε ένα array από DataPage instances. Λαμβάνει ορίσματα: το byte array (data) τον αριθμό των εγγραφών προς μετατροπή (numberOfRecords) και το μέγεθος της κάθε εγγραφής (capacityOfRecords), τα δύο τελευταία είναι ακέραιοι. Επιστρέφει ένα DataPage array.
- **getAlphaNumericString**: Παράγει τυχαία αλφαριθμητικά. Παίρνει όρισμα το μέγεθος του αλφαριθμητικού και επιστρέφει ένα (ψευδο)τυχαίο String.
- **binarySearchFile()**: Κάνει την δυαδική αναζήτηση στο αρχείο. Όρισματα: το όνομα του αρχείου (String file), το αριστερό όριο της αναζήτησης (int l), το δεξί όριο της αναζήτησης (int r), το κλειδί το για το οποίο γίνεται η αναζήτηση (int x) και τον αριθμό των προσβάσεων στον δίσκο (int diskAccesses) μέχρι την εκάσθωτε κλήση της συνάρτησης (γίνεται χρήση αναδρομής). Επιστρέφει ένα array με δύο ακεραίους: ο πρώτος είναι ο αριθμός της σελίδας στην οποία βρέθηκε το κλειδί και ο δεύτερος ο αριθμός των προσβάσεων στον δίσκο της αναζήτησης (χρειάζεται για την ανάλυση της πολυπλοκότητας).

3 Πηγές

- https://www.geeksforgeeks.org/generate-random-string-of-given-size-in-java/ (method 1)
- Γενικές πληροφορίες για την κλάση ByteBuffer: https://docs.oracle.com/javase/7/docs/api/java/nio/ByteBuffer.html
- Πληροφορίες για την συνάρτηση arraycopy() από την κλάση System: https://docs.oracle.com/javase/7/docs/api/java/lang/System.htmlarraycopy(java.lang.Object%20int,%20java.lang.Object,%2
- Πληροφορίες για την java.util.Arrays και συγκεκριμένα για τις μεθόδους sort() και copyOfRange(): https://docs.oracle.com/javase/7/docs/api/java/util/Arrays.html