

## **K08 Δομές Δεδομένων και Τεχνικές Προγραμματισμού**

**Διδάσκων: Μανόλης Κουμπάρκης**

**Εαρινό Εξάμηνο 2020-2021**

### **Εργασία 2**

**Ανακοινώθηκε στις 12 Απριλίου 2018**

**Προθεσμία: 9 Μαΐου 2021 στις 23:59**

**20% του συνολικού βαθμού στο μάθημα (Άριστα=285 μονάδες, υπάρχουν επίσης 50 μονάδες bonus)**

**Προσοχή:** Πριν διαβάσετε παρακάτω, διαβάστε παρακαλώ προσεκτικά τις οδηγίες υποβολής των ασκήσεων που βρίσκονται στην ιστοσελίδα <http://cgi.di.uoa.gr/~k08/homework.html>, ειδικά ότι αναφέρεται στο github και τα σχετικά αρχεία.

**Απορίες:** Αν έχετε απορίες σχετικά με την εργασία, ρωτήστε στο piazza και όχι στέλνοντας e-mail στον διδάσκοντα. Τέτοια e-mails ΔΕΝ θα λαμβάνουν απάντηση.

**Κώδικας:** Ο κώδικας που παρουσιάσαμε στις διαλέξεις του μαθήματος (Ενότητες 6-12) και θα χρειαστείτε για την εργασία αυτή βρίσκεται στο παρακάτω private repository του classroom του μαθήματος: <https://github.com/artioi-k08/2021-ergasia-2>. Για να συνδεθείτε στο repository αυτό, θα πρέπει να χρησιμοποιήσετε το λινκ <https://classroom.github.com/a/zYLBLGDk>.

Όταν συνδεθείτε θα μπορείτε να δείτε το προσωπικό σας repository <https://github.com/artioi-k08/2021-ergasia-2-<github-your-username>> στο οποίο θα δουλέψετε για την Εργασία 2.

Μετά τη σύνδεση σας, στο προσωπικό σας repository, θα βρείτε τον κώδικα που καλύπτει τις Ενότητες 6-12 του μαθήματος οργανωμένο σε κατάλληλους φακέλους. Θα βρείτε επίσης και ένα φάκελο **solutions-ergasia2** με υπο-φακέλους **question1, question2, ..., question11** στους οποίους θα πρέπει να γράψετε τον

κώδικα ή τις απαντήσεις σας για τα 11 ερωτήματα της εργασίας που θα βρείτε παρακάτω. Παρακαλώ τηρείστε ευλαβικά αυτήν την οργάνωση αλλιώς θα χάσετε 20% του συνολικού βαθμού κατά την βαθμολόγηση. Για τα θεωρητικά ερωτήματα, οι απαντήσεις πρέπει να είναι σε ένα αρχείο τύπου pdf.

**Κύριο πρόγραμμα:** Σε όλες τις παρακάτω προγραμματιστικές ασκήσεις θα πρέπει να υλοποιήσετε και ένα κύριο πρόγραμμα (συνάρτηση `main`) το οποίο θα διαβάζει τα δεδομένα εισόδου, θα επιδεικνύει τη λειτουργικότητα της συνάρτησης σας για κατάλληλα επιλεγμένες εισόδους, και θα πείθει τον βαθμολογητή ώστε να σας βαθμολογήσει με τον υψηλότερο δυνατό βαθμό.

1. Θεωρήστε το πρόγραμμα που βρίσκεται στο αρχείο `insertion-sort.c` που βρίσκεται στο repository σας (φάκελος `insertion-sort-code`). Το πρόγραμμα αυτό ταξινομεί ένα πίνακα ακεραίων με τη μέθοδο της ταξινόμησης με εισαγωγή. Ποια είναι η χρονική υπολογιστική πολυπλοκότητα χειρότερης περίπτωσης της συνάρτησης `sort`; Ποια είναι η αντίστοιχη πολυπλοκότητα της `main`; Υποθέστε ότι η κλήση `rand()` εκτελείται σε σταθερό χρόνο. Να εξηγήσετε με λεπτομέρεια τις απαντήσεις σας.

(5 μονάδες)

2. Υποθέστε ότι έχουμε 10 αλγόριθμους A, B, Γ, Δ, E, Z, H, Θ, I και K με τις παρακάτω υπολογιστικές πολυπλοκότητες χρόνου (παραλείπουμε το  $O()$ ).

A.  $2000n$     B.  $50n + \log n$     Γ.  $2^{3000n \log n}$     Δ.  $2^{300 \log n}$     E.  $n^4$   
Z.  $6n^4 + n$     H.  $n \log n 2^{n+5}$     Θ.  $2^{2^n}$     I.  $2^{2^{\log n}}$     K.  $2^{n+\log n}$

Να ταξινομήσετε τους αλγόριθμους από τον καλύτερο στον χειρότερο με βάση την υπολογιστική πολυπλοκότητα τους. Να δώσετε λεπτομερώς όσους μαθηματικούς υπολογισμούς χρειάζονται για να τεκμηριώσετε την απάντησή σας.

(5 μονάδες)

3. Να οργανώσετε τα γράμματα της λέξης ΟΛΥΜΠΙΑΚΟΣ σε ένα αρχικά κενό σωρό μεγίστων (`max heap`) χρησιμοποιώντας τον αλγόριθμο εισαγωγής που δίνεται στην σελίδα 64-76 των διαφανειών της Ενότητας 8 από την ιστοσελίδα του μαθήματος. Τα βήματα του αλγόριθμου πρέπει να παρουσιαστούν λεπτομερώς χρησιμοποιώντας την δενδρική αναπαράσταση του σωρού όπως

κάνουμε στα παραδείγματα των σελίδων αυτών. Δεν χρειάζεται να δείχνετε τον αντίστοιχο πίνακα.

**(10 μονάδες)**

4. Σχεδιάστε το δυαδικό δένδρο αναζήτησης που προκύπτει όταν εισάγουμε τα κλειδιά N E X T σε ένα αρχικά άδειο δένδρο χρησιμοποιώντας τον αλγόριθμο της σελίδας 36 της Ενότητας 9 των διαλέξεων. Ομοίως τα κλειδιά W E D N E S D A Y σε ένα δεύτερο δυαδικό δένδρο αναζήτησης. Τέλος, συνενώστε τα δύο δένδρα εκτελώντας την πράξη join που μελετήσαμε στην ίδια ενότητα διαλέξεων.

**(10 μονάδες)**

5. Δώστε μια σειρά εισαγωγής για τα κλειδιά E A S Y Q U E S T I O N σε ένα αρχικά άδειο δυαδικό δένδρο αναζήτησης τέτοια ώστε το δένδρο που παράγεται είναι ισοδύναμο με δυαδική αναζήτηση κατά την εξής έννοια: μια ακολουθία συγκρίσεων κατά την αναζήτηση ενός κλειδιού στο δένδρο είναι η ίδια με την ακολουθία συγκρίσεων που χρησιμοποιείται από την δυαδική αναζήτηση για το ίδιο σύνολο κλειδιών.

**(10 μονάδες)**

6. Δώστε μια ακολουθία 10 κλειδιών (χρησιμοποιήστε τα γράμματα A έως K) η οποία, όταν τα κλειδιά εισάγονται με τη σειρά αυτή σε ένα αρχικά άδειο δυαδικό δένδρο αναζήτησης με τη μέθοδο της εισαγωγής στη ρίζα (σελίδες 81-90 των διαφανειών της Ενότητας 9), απαιτείται μέγιστος αριθμός συγκρίσεων για να χτιστεί το δένδρο. Να δώσετε αυτό τον αριθμό συγκρίσεων.

**(10 μονάδες)**

7. Στην άσκηση αυτή θα επεκτείνετε την υλοποίηση του αφαιρετικού τύπου δεδομένων Πίνακας Συμβόλων (Symbol Table) με δένδρα δυαδικής αναζήτησης η οποία παρουσιάζεται στις διαφάνειες της Ενότητας 9.

7.1. Να οργανώσετε τον κώδικα του αρχείου `symbol-table.txt` που βρίσκεται στο repository σας (φάκελος `symbol-table-code`) σε ένα module της C το οποίο υλοποιεί τον αφαιρετικό τύπο δεδομένων Πίνακας Συμβόλων με το interface που δίνεται στη σελίδα 5 των σχετικών διαφανειών. Μπορείτε να χρησιμοποιήσετε

και να επεκτείνετε τη συνάρτηση `main` της σελίδας 65 των διαφανειών για να επιδείξετε τη λειτουργία του πίνακα συμβόλων με κατάλληλες κλήσεις στις συναρτήσεις που προσφέρονται από το `interface`. Είναι δική σας ευθύνη να επεκτείνετε τον κώδικα που σας δίνεται, ώστε να δουλεύουν πλήρως οι διάφορες συναρτήσεις. Για παράδειγμα, θα χρειαστεί να επεκτείνετε κάποιες συναρτήσεις που δεν ενημερώνουν το πεδίο `N` του `struct STnode` ώστε να το κάνουν.

- 7.2. Να προσθέσετε στο `interface` και να υλοποιήσετε τη συνάρτηση `void STdeleteAll(Item)` που θα είναι μια νέα έκδοση της συνάρτησης `STdelete` (σελίδα 108 των διαφανειών της Ενότητας 9) η οποία διαγράφει όλα τα στοιχεία του πίνακα συμβόλων (δηλαδή, όλους τους κόμβους του δένδρου) που έχουν κλειδιά ίσα με το δοσμένο κλειδί. Να προσθέσετε κώδικα στη `main` η οποία να δείχνει τη λειτουργικότητα της συνάρτησης σας.
- 7.3. Να επεκτείνετε το λογισμικό που αναπτύξατε στα ερωτήματα 7.1 και 7.2 ώστε να μπορεί να χρησιμοποιηθεί στην περίπτωση που θέλουμε ο πίνακας συμβόλων να λειτουργεί σαν μητρώο των φοιτητών του τμήματος μας. Υποθέστε ότι σε κάθε `item` τώρα δεν αποθηκεύουμε ένα ακέραιο αλλά τις εξής πληροφορίες: το κλειδί που είναι ο αριθμός μητρώου του φοιτητή, το όνομα του και το επώνυμο του.
- 7.4. Παρατηρήστε ότι ο κώδικας που δώσαμε στο μάθημα, και τον οποίο χρησιμοποιήσατε και επεκτείνατε στα ερωτήματα 7.1, 7.2 και 7.3, μας επιτρέπει να έχουμε ένα μόνο πίνακα συμβόλων. Να υλοποιήσετε μια νέα έκδοση του λογισμικού των 7.1 και 7.2 που να επιτρέπει τη δημιουργία πολλών πινάκων συμβόλων και την χρήση τους από ένα κύριο πρόγραμμα που θα επιδεικνύει τη λειτουργικότητα του λογισμικού σας.

Σε όλα τα παραπάνω ερωτήματα, θα πρέπει ο κώδικας των `modules` που θα αναπτύξετε να κάνει την καλύτερη δυνατή απόκρυψη πληροφορίας.

**(20+10+10+50=90 μονάδες, οι 50 μονάδες είναι για το ερώτημα 7.4 και είναι bonus)**

8. Να σχεδιάσετε το δένδρο AVL που προκύπτει όταν τα κλειδιά 1, 2, 3, 5, 6, 4, 7, 8, 10, 9 εισάγονται ένα προς ένα με τη σειρά που δίνονται σε ένα αρχικά κενό δένδρο. Μετά να διαγράψετε το κλειδί που βρίσκεται στη ρίζα του δένδρου αυτού. Όλα τα βήματα των αλγορίθμων εισαγωγής και διαγραφής πρέπει να δίνονται αναλυτικά.

**(15 μονάδες)**

9. Να σχεδιάσετε το δένδρο (2,4) που προκύπτει όταν η ακολουθία κλειδιών 5 1 22 25 20 21 15 18 19 2 30 27 40 εισάγεται σε ένα αρχικά κενό δένδρο. Μετά να σχεδιάσετε το δένδρο (2,4) που προκύπτει αν διαγραφούν όλα τα κλειδιά που βρίσκονται στη ρίζα από το μικρότερο προς το μεγαλύτερο. Όλα τα βήματα των αλγορίθμων εισαγωγής και διαγραφής πρέπει να δίνονται αναλυτικά.

**(15 μονάδες)**

10. Να σχεδιάσετε το κόκκινο-μαύρο δένδρο που προκύπτει αν εισάγουμε τα κλειδιά 5 1 19 17 21 20 9 15 14 2 12 11 σε ένα αρχικά κενό δένδρο. Μετά να σχεδιάσετε το κόκκινο-μαύρο δένδρο που θα προκύψει αν διαγράψουμε το κλειδί που βρίσκεται στη ρίζα.

**(15 μονάδες)**

11. Να χρησιμοποιήσετε δένδρα (2,4) για να υλοποιήσετε τη λειτουργικότητα του πίνακα συμβόλων από το παραπάνω ερώτημα 7.1. Θα χρειαστεί να υλοποιήσετε κατάλληλες συναρτήσεις όπως αυτές του ερωτήματος 7.1, που όμως τώρα θα χρησιμοποιούν τους αλγόριθμους αναζήτησης, εισαγωγής και διαγραφής κλειδιού που παρουσιάζονται στις διαφάνειες της Ενότητας 11. Για τις λειτουργίες του πίνακα συμβόλων που δεν έχουμε δώσει αντίστοιχους αλγόριθμους που δουλεύουν σε δένδρα (2,4) (π.χ., βρες το στοιχείο με το  $k$ -οστό μικρότερο κλειδί), θα πρέπει να τους σχεδιάσετε και να τους υλοποιήσετε εσείς.

**(150 μονάδες)**

**Καλή Επιτυχία!**