

6^ο Εργαστήριο:

Ταξινόμηση 2 Quick sort, Merge sort

Ημερομηνία παράδοσης: Κυριακή 18 Απριλίου

Μαθησιακοί στόχοι:

- Ταξινόμηση αντικειμένων

Χρησιμοποιώντας τον πίνακα των φοιτητών που έχετε δημιουργήσει στο προηγούμενο εργαστήριο να υλοποιήσετε τις ακόλουθες μεθόδους ταξινόμησης.

Ασκηση 1: Ταξινόμηση Quick sort

- (α) Χρησιμοποιώντας μόνο getter μεθόδους, να εμφανίσετε για όλους τους φοιτητές, όλα τα πεδία της κλάσης
- (β) Να ταξινομήσετε τους φοιτητές με κλειδί ταξινόμησης το [έτος εισαγωγής](#) τους
- (γ) Να εμφανίσετε ταξινομημένα τους φοιτητές

Ασκηση 2: Ταξινόμηση Merge sort

- (α) Χρησιμοποιώντας μόνο getter μεθόδους, να εμφανίσετε για όλους τους φοιτητές, όλα τα πεδία της κλάσης
- (β) Να ταξινομήσετε τους φοιτητές με κλειδί ταξινόμησης το [ονοματεπώνυμό](#) τους
- (γ) Να εμφανίσετε ταξινομημένα όλους τους φοιτητές

Οδηγίες

- Στην περίπτωση αντιμετάθεσης αντικειμένων – **swap** – μετακινούμε τους **δείκτες** των αντικειμένων (θέσεις πίνακα) και **όχι των πεδίων**

Σύνδεσμοι

Arrays.toString(): <https://www.geeksforgeeks.org/arrays-tostring-in-java-with-examples/>

Java.lang.System.arraycopy(): https://www.tutorialspoint.com/java/lang/system_arraycopy.htm

Quick sort: <https://www.javatpoint.com/daa-quick-sort>

<https://www.softwaretestinghelp.com/quicksort-in-java/>

https://www.youtube.com/watch?v=HDQd6_OTJIE

Merge sort: <https://howtodoinjava.com/algorithm/merge-sort-java-example/>

Ψευδοκώδικας

Following is the algorithm for the technique.

#1) Declare an array myArray of length N

#2) Check if $N=1$, myArray is already sorted

#3) If N is greater than 1,

- set left = 0, right = N-1
- compute middle = $(\text{left} + \text{right})/2$
- Call subroutine **merge_sort** (myArray, left, middle) => this sorts first half of the array
- Call subroutine **merge_sort** (myArray, middle+1, right) => this will sort the second half of the array
- Call subroutine **merge** (myArray, left, middle, right) to merge arrays sorted in the above steps.

#4) Exit

