

Τροποποίηση του αλγορίθμου σειριακής αναζήτησης ώστε να λειτουργεί βέλτιστα σε ταξινομημένο πίνακα

Αλγόριθμος Sequential_Search_Sorted_Table

Δεδομένα // $n, table, key$ //

$done \leftarrow \text{ψευδής}$

$position \leftarrow 0$

$i \leftarrow 1$

Όσο ($done = \text{ψευδής}$) **και** ($i \leq n$) **επανάλαβε** ! για ταξινομημένο πίνακα με αύξουσα διάταξη

Αν ($table[i] > key$) **τότε** ! σταμάτα την επανάληψη, δεν θα βρεθεί το στοιχείο

$done \leftarrow \text{αληθής}$

Αλλιώς **Αν** ($table[i] = key$) **τότε** ! σταμάτα την επανάληψη, το στοιχείο βρέθηκε

$done \leftarrow \text{αληθής}$

$position \leftarrow i$

Αλλιώς ! συνέχισε την την επανάληψη

$i \leftarrow i + 1$

Τέλος_αν

Τέλος_επανάληψης

Αν ($position \neq 0$) **τότε**

Εκτύπωσε "Το στοιχείο ", key , " ευρέθη στη θέση ", $position$

Αλλιώς

Εκτύπωσε "Το στοιχείο ", key , " δεν ευρέθη στον δοθέντα πίνακα"

Τέλος_αν

Αποτελέσματα // $position$ //

Τέλος Sequential_Search_Sorted_Table

Παρατηρήσεις:

Η αναζήτηση σταματά όταν η τιμή προς αναζήτηση (μεταβλητή key) είναι μικρότερη από την τρέχουσα τιμή του πίνακα $table$

Σε αυτήν την περίπτωση δεν έχει νόημα να συνεχιστεί η αναζήτηση καθώς δεν αναμένεται να έχει επιτυχές αποτέλεσμα

4	9	12	19	23	45	53	67
---	---	----	----	----	----	----	----

Για παράδειγμα η αναζήτηση για την τιμή 14 στον παραπάνω πίνακα δεν έχει νόημα να συνεχιστεί μετά την 4^η επανάληψη ($i=4, table[i]=19$) καθώς ο αριθμός 14 που είναι μικρότερος του 19 αποκλείεται να βρίσκεται σε κάποια από τις επόμενες θέσεις του πίνακα. Έτσι, αποφεύγονται άσκοπες επαναλήψεις. Το αν εντοπίστηκε ή όχι η τιμή key το ανιχνεύουμε με τη μεταβλητή $position$ και όχι με την $done$ καθώς η τελευταία λαμβάνει την τιμή αληθής ακόμη κι αν δεν ευρέθη η key