

Παράδοση έως

17/11/2020

ΗΡΑΚΛΕΙΟ ΚΡΗΤΗΣ 2020

ΤΜΗΜΑ ΗΛΕΚΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΑΛΓΟΡΙΘΜΟΙ (ΤΠ30Κ4)

3η Εργαστηριακή άσκηση

ΜΕΛΑΚΗΣ ΝΙΚΟΛΑΟΣ ΤΠ4726

1. Υλοποιήστε σαν συνάρτηση τον αλγόριθμο δυαδικής αναζήτησης (τον επαναληπτικό και ΟΧΙ τον αναδρομικό).

**int binarySearch (int pin[], int l, int r, int num)**

Κώδικας: <BinarySearch.cpp>

2. Υλοποιήστε σαν συνάρτηση τον αλγόριθμο αναζήτησης παρεμβολής.

**int interpolationSearch (int pin[], int low, int high, int num)**

Κώδικας: <InterpolationSearch.cpp>

3. Συγκρίνετε τον αριθμό των βημάτων που εκτελεί ο καθένας από αυτούς τους αλγορίθμους για να εντοπίσει τη θέση στοιχείου το οποίο υπάρχει ήδη σε πίνακα με τυχαίες (random) τιμές οι οποίες είναι σε αύξουσα διάταξη.

Όταν λέμε αριθμό βημάτων εννοούμε τον αριθμό των επαναλήψεων που εκτελεί ο βασικός βρόγχος (εξωτερικό while loop) του κάθε αλγορίθμου.

Το στοιχείο που θα αναζητάει ο αλγόριθμος θα πρέπει να είναι ήδη μέσα στον πίνακα. Δηλαδή όταν καλείτε τις συναρτήσεις binarySearch και interpolationSearch, στο όρισμα num θα περνάτε ένα τυχαίο στοιχείο που υπάρχει ήδη στον πίνακα.

Ένα κομμάτι κώδικα με το οποίο μπορείτε να γεμίσετε έναν πίνακα με τυχαίες τιμές σε αύξουσα διάταξη είναι:

int pin[N], i;

/\* Initialize array \*/

srand(time(NULL));

pin[0] = rand()%100;

for(i=1;i<N;i++)

pin[i] = pin[i-1] + rand()%10;

Αποτελέσματα(Ν=100)

Για πίνακα 100 ακεραίων.(δυαδική αναζήτηση)

1ο execute χρειάστηκαν 6 βήματα για την εύρεση του στοιχείου.

2ο execute χρειάστηκαν 7 βήματα για την εύρεση του στοιχείου.

3ο execute χρειάστηκαν 4 βήματα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 7 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκαν 3 βήματα για την εύρεση του στοιχείου.

Αποτελέσματα(Ν=100)

Για πίνακα 100 ακεραίων.(αλγόριθμος παρεμβολής)

1ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

2ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

3ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

Αποτελέσματα(Ν=1000)

Για πίνακα 1000 ακεραίων. (δυαδική αναζήτηση)

1ο execute χρειάστηκαν 9 βήματα για την εύρεση του στοιχείου.

2ο execute χρειάστηκαν 10 βήματα για την εύρεση του στοιχείου.

3ο execute χρειάστηκαν 9 βήματα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 10 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκαν 8 βήματα για την εύρεση του στοιχείου.

Αποτελέσματα(Ν=1000)

Για πίνακα 1000 ακεραίων.(αλγόριθμος παρεμβολής)

1ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

2ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

3ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

Αποτελέσματα(Ν=5000)

Για πίνακα 5000 ακεραίων. (δυαδική αναζήτηση)

1ο execute χρειάστηκαν 12 βήματα για την εύρεση του στοιχείου.

2ο execute χρειάστηκαν 11 βήματα για την εύρεση του στοιχείου.

3ο execute χρειάστηκαν 12 βήματα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 10 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκαν 13 βήματα για την εύρεση του στοιχείου.

Αποτελέσματα(Ν=5000)

Για πίνακα 5000 ακεραίων.(αλγόριθμος παρεμβολής)

1ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

2ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

3ο execute χρειάστηκε 1 βήμα για την εύρεση του στοιχείου.

4ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

5ο execute χρειάστηκαν 2 βήματα για την εύρεση του στοιχείου.

**Θα πρέπει να παραδώσετε:** (έως 17/11/20)

- Τους κώδικες των δύο αλγορίθμων

- Μία μικρή αναφορά όπου θα αναφέρετε τον αριθμό βημάτων για κάθε αλγόριθμο και για διαφορετικά πλήθη των στοιχείων εισόδου, δηλαδή για 100, 1000 και 5000 στοιχεία στον πίνακα. Για κάθε διαφορετικό πλήθος εισόδου θα πρέπει να τρέξετε τον κάθε αλγόριθμο πέντε φορές.