Αλγόριθμοι και προχωρημένες δομές δεδομένων
ΣΕΤ ΑΣΚΗΣΕΩΝ #1

Ημερομηνία παράδοσης 23 Νοεμβρίου 2018

Χρυσόστομος Στύλιος και Χρήστος Γκόγκος

9/11/2018

Οι ακόλουθες ασκήσεις θα πρέπει να υλοποιηθούν σε C++, χρησιμοποιώντας σύγχρονες δυνατότητες της γλώσσας. Ανεβάστε τον κώδικα και τα αποτελέσματα εκτέλεσης σε αποθετήριο (git) που θα δημιουργήσετε για το σκοπό αυτό και το οποίο θα μπορεί να είναι προσβάσιμο και από τρίτους. Για τη δημιουργία του αποθετηρίου χρησιμοποιήστε το GitHub ή το BitBucket.

Άσκηση 1

Δημιουργήστε έναν δωρεάν λογαριασμό στο Kattis https://open.kattis.com/. Επιλύστε τα προβλήματα με κωδικούς: hello, faktor, everywhere και veci.

1. https://open.kattis.com/problems/hello
2. https://open.kattis.com/problems/faktor
3. https://open.kattis.com/problems/everywhere
4. https://open.kattis.com/problems/veci

Άσκηση 2

Στο πρόβλημα αυτό δίνεται ένας πίνακα με ακέραιες θετικές και αρνητικές τιμές και ζητείται να βρεθεί η υποακολουθία συνεχόμενων τιμών του πίνακα με το μεγαλύτερο άθροισμα. Για παράδειγμα αν ο πίνακας έχει δηλωθεί και αρχικοποιηθεί ως:

```
int a[]={2,6,-3,5,1,2,1,-2,6};
```

το άθροισμα της μεγαλύτερης υποακολουθίας είναι 9 = 5+1+2+1
Δώστε τέσσερις λύσεις που να έχουν χρονική πολυπλοκότητα $O(n^3)$, $O(n^2)$, $O(n \log n)$ και $O(n)$ αντίστοιχα. Για τη λύση $O(n \log n)$ συμβουλευτείτε το https://www.geeksforgeeks.org/maximum-subarray-sum-using-divide-and-conquer-algorithm/, ενώ για την λύση $O(n)$ εντοπίστε στο διαδίκτυο πληροφορίες για τον αλγόριθμο του Kadane. Κατασκευάστε συναρτήσεις με τις ακόλουθες υπογραφές.

```
int max_subarray1(int a[], int n)
int max_subarray2(int a[], int n)
int max_subarray3(int a[], int n)
int max_subarray4(int a[], int n)
```

Δοκιμάστε τις λύσεις σας για πίνακα 10000 τυχαίων ακέραιων τιμών στο διάστημα [-1000,1000]. Χρονομετρήστε τον κώδικα.

Άσκηση 3

(προαιρετική)
Ο τυπικός αλγόριθμος πολλαπλασιασμού δύο δισδιάστατων πινάκων A[n,n], B[n,n] έχει χρονική πολυπλοκότητα $O(n^3)$. Ωστόσο, υπάρχει και ο αλγόριθμος του Strassen ο οποίος έχει χρονική πολυπλοκότητα $O(n^{2.807})$. Ζητείται η υλοποίηση του αλγορίθμου του Strassen για τον πολλαπλασιασμό δισδιάστατων πινάκων και η εμπειρική σύγκριση του χρόνου εκτέλεσής του με τον τυπικό αλγόριθμο πολλαπλασιασμού δισδιάστατων πινάκων.