

HY240: Δομές Δεδομένων

Εαρινό Εξάμηνο – Ακαδημαϊκό Έτος 2016-17

Παναγιώτα Φατούρου

1^η Σειρά Ασκήσεων

Ημερομηνία Παράδοσης: Τρίτη, 28 Φεβρουαρίου 2017

Τρόπος Παράδοσης: Οι ασκήσεις μπορούν να παραδοθούν είτε σε ηλεκτρονική μορφή χρησιμοποιώντας το πρόγραμμα turnin (δείτε οδηγίες στην ιστοσελίδα του μαθήματος), είτε στους βοηθούς του μαθήματος, την Τρίτη, 28 Φεβρουαρίου 2017, ώρα 13:00-14:00. Ασκήσεις που παραδίδονται μετά τις 14:00 της Τρίτης, 28/02/2017 θεωρούνται εκπρόθεσμες. Εκπρόθεσμες ασκήσεις γίνονται δεκτές σε ηλεκτρονική μορφή και η αποστολή τους πρέπει να γίνει χρησιμοποιώντας το πρόγραμμα turnin. **Ασκήσεις μέσω ηλεκτρονικού ταχυδρομείου δεν γίνονται δεκτές.**

Άσκηση 1 [20 Μονάδες]

Έστω $T_1(n)$ το συνολικό πλήθος των φορών που εκτελείται η εντολή $x = x + 1$ από τη ρουτίνα Tricky1() και $T_2(n)$ το συνολικό πλήθος των φορών που εκτελείται η εντολή $x = x + 1$ από τη ρουτίνα Tricky2(). Υπολογίστε τα $T_1(n)$ και $T_2(n)$. Μελετήστε την τάξη (συμβολισμός O) των $T_1(n)$ και $T_2(n)$.

<pre> procedure Tricky1(int n) { for i = 1 to $\sqrt{n}$ do for j = n^2 down to i do for k = 1 to $\sqrt{n}$ do $x = x + 1$; } }</pre>	<pre> procedure Tricky2(int n) { for i = 1 to n do for j = 1 to i do for (k = 0; k < n^2; k+ = n) $x = x + 1$; } }</pre>
--	---

Άσκηση 2 [30 μονάδες]

α. Ποιες από τις ακόλουθες προτάσεις είναι αληθείς; Θεωρήστε ότι $f = f(n)$ και $g = g(n)$ είναι συναρτήσεις με πεδίο ορισμού το σύνολο των θετικών αριθμών. [20M]

Πρόταση	Αληθής ή Ψευδής?
$n^2 + 2n \log^2 n + n + 1 = \Theta(n^3)$	
$n^2 \log^2 n + n \log n + n = O(n^3 \log^2 n)$	
$\sum_{i=1}^n (5 \cdot (i + 2)) = \Omega(n^2)$	

Αποδείξτε τους ισχυρισμούς σας.

β. Αποδείξτε την αναστροφική ιδιότητα των O, Ω .

[5M]

γ. Αποδείξτε ότι αν $f(n) = O(g(n))$ και $h(n) = O(g(n))$, τότε $(c_1 f + c_2 h)(n) = O(g(n))$, για οποιαδήποτε σταθερές $c_1, c_2 \in \mathbb{R}^+$.

[5M]

Άσκηση 3 [30 Μονάδες]

α. Δίνεται η παρακάτω συνάρτηση (Αιγυπτιακός πολλαπλασιασμός):

$$m(x,y) = \begin{cases} 0, & \text{αν } y = 0 \\ m(x+x, y/2), & \text{αν } y \text{ άρτιος \& } \neq 0 \\ x + m(x, y-1), & \text{αν } y \text{ περιττός \& } \neq 0 \end{cases}$$

(i) Παρουσιάστε ψευδοκώδικα για μια αναδρομική συνάρτηση

int EgyptianMultiplication(int x, int y)

η οποία θα πραγματοποιεί αιγυπτιακό πολλαπλασιασμό σύμφωνα με την παραπάνω αναδρομική σχέση. [7M]

(ii) Ιχνηλατίστε την εκτέλεση της EgyptianMultiplication(10,22) (δηλαδή παρουσιάστε αναλυτικά την εκτέλεσή της, καθώς και τα activation records στη μνήμη, όταν καλείτε με $x = 10$ και $y = 22$). [10M]

β. Ας υποθέσουμε πως πρέπει να πληρώσετε ένα πρόστιμο των $n > 8$ ευρώ στην τροχαία. Το ποσό αυτό πρέπει να πληρωθεί αγοράζοντας παράβολα, αλλά τη συγκεκριμένη μέρα έχουν μείνει μόνο παράβολα των 3 και των 5 ευρώ. Αποδείξτε επαγωγικά πως $\forall n \geq 8$, είναι δυνατόν να πληρωθεί το ποσό των n ευρώ αγοράζοντας παράβολα των 3 ευρώ και των 5 ευρώ που παρέχονται τη συγκεκριμένη μέρα. [10M]

Παρατήρηση: Εξετάστε προσεκτικά τα βήματα της επαγωγής για να δείτε αν η απόδειξη που παρέχετε είναι σωστή. Προσοχή στις συνθήκες που ισχύουν για την επαγωγική υπόθεση.

Άσκηση 4 [20 Μονάδες]

Θεωρήστε την ακόλουθη διαδικασία, όπου η swap(int a, int b) ανταλλάζει την τιμή της μεταβλητής a με την τιμή της μεταβλητής b.

```
void Alg(table T[0..n-1]: array of integers) {
    for m = 0 to n-2 do {
        j = m;
        for k = m+1 to n-1 do {
            if T[k] < T[j] then j = k;
        }
        swap(T[m], T[j]);
    }
}
```

α. Παρουσιάστε σύντομη περιγραφή του τρόπου λειτουργίας του Alg(). Ποιο είναι το αποτέλεσμα του Alg(); [5%]

β. Ιχνηλατίστε (δηλαδή «εκτελέστε» στο χαρτί - κάνετε trace) το Alg(T[0..6]) για την περίπτωση που $T = [7, 4, 5, 3, 1, 2]$. [7%]

γ. Μελετήστε την τάξη της χρονικής πολυπλοκότητας της Alg(). [10%]

Υπόδειξη: Υποθέστε πως η χρονική πολυπλοκότητα της swap() είναι $O(1)$.