

HY240: Δομές Δεδομένων
Χειμερινό Εξάμηνο – Ακαδημαϊκό Έτος 2008-09
Παναγιώτα Φατούρου

1^ο Σετ Ασκήσεων

Ημερομηνία Παράδοσης: Παρασκευή, 10 Οκτωβρίου 2008

Τρόπος Παράδοσης: Στο εργαστήριο των μεταπτυχιακών στο υπόγειο των λευκών κτιρίων, ώρα 12:00-13:00. Ασκήσεις που παραδίδονται μετά τις 13:00 θεωρούνται εκπρόθεσμες. Οι φοιτητές μπορούν να παραδίδουν τις ασκήσεις τους και νωρίτερα από αυτήν την ημερομηνία αλλά δικαιολογίες όπως «ήρθα την τάδε μέρα και δεν ήσασταν στο γραφείο σας» δεν θα γίνουν δεκτές. Όσοι από τους φοιτητές δεν έχουν καταφέρει να παραδώσουν τις ασκήσεις τους πριν τις 10/10 θα πρέπει να τις παραδώσουν στο εργαστήριο των μεταπτυχιακών την προκαθορισμένη μέρα και ώρα. Οι φοιτητές μπορούν να παραδίδουν τις ασκήσεις τους ηλεκτρονικά αλλά σε αυτή την περίπτωση απαιτείται να γίνουν scan οι λύσεις και να σταλούν μέσω e-mail στο λογαριασμό hy240@csd.uoc.gr. Αυτός είναι και ο μοναδικός τρόπος παράδοσης εκπρόθεσμων ασκήσεων.

Θεωρητικό Μέρος

1. Θεωρήστε την ακολουθία αριθμών Fibonacci που ορίζεται ως εξής:

$$F(n) = \begin{cases} 0 & \text{αν } n = 0 \\ 1 & \text{αν } n = 1 \\ F(n-1) + F(n-2) & \text{αν } n > 1 \end{cases}$$

Η αναδρομική υλοποίηση της συνάρτησης αυτής παρουσιάζεται στη συνέχεια:

```
integer Fib(integer n) {  
    int n1, n2;  
    if (n < 2) then return n;  
    else {  
        n1 = Fib(n-1);  
        n2 = Fib(n-2);  
        return (n1+n2);  
    }  
}
```

Ζητείται να ιχνηλατηθεί στο χαρτί (δηλαδή να γίνει trace) η Fib(n) για την περίπτωση που $n = 5$. Πρέπει να παρουσιαστούν όλες οι αναδρομικές κλήσεις της Fib καθώς και οι τιμές των μεταβλητών n , $n1$ και $n2$ για κάθε εκτέλεση της Fib. Πρέπει επίσης να παρουσιαστεί ο χώρος στη μνήμη που κατανέμεται για την εκτέλεση των αναδρομικών κλήσεων της Fib.

2. Να παρουσιαστούν δύο αλγόριθμοι υπολογισμού του $\sum_{i=1}^n a_i x^i$, $1 \leq i \leq n$, οι οποίοι να έχουν χρονική πολυπλοκότητα διαφορετικής τάξης. Ο κάθε αλγόριθμος θα παίρνει ως παράμετρο το x και θα επιστρέφει την τιμή του πολυωνύμου $\sum_{i=1}^n a_i x^i$, $1 \leq i \leq n$, για την δεδομένη τιμή του x . Να θεωρηθεί ότι τα a_i είναι γνωστές σταθερές. Παρουσιάστε ψευδοκώδικα για τους δύο αλγορίθμους και μελετήστε φορμαλιστικά την τάξη της χρονικής τους πολυπλοκότητας (βάσει των συμβολισμών O , Ω και Θ).

3. α. Αποδείξτε τη μεταβατική και τη συμμετρική ιδιότητα του Θ.

β. Αποδείξτε ότι $\lg(n!) = O(n \lg n)$.

γ. Ποια από τα ακόλουθα είναι αληθή και γιατί;

i. $n^3 + 4n^2 + 5n + 10 = \Theta(n^3)$

ii. $\lg(n^3) = \Theta(n \lg n)$

iii. $\log(\sqrt{n}) = \Theta(\lg n)$

iv. $\sum_{i=1}^n (2 * (i+1)) = O(n^3)$

v. $\min(700, n^2) = \Theta(1)$

4. α. Αποδείξτε επαγωγικά ότι αν $T(0) = 0$ και $T(n) = 2T(n-1) + 1$, $n > 0$, τότε $T(n) = 2^n - 1$.

β. Θεωρήστε τη συνάρτηση $f: \mathbb{N} \rightarrow \mathbb{N}$ που ορίζεται ως εξής:

$$f(0) = 1, f(1) = 2 \text{ και } f(n) = 4f(n-2) + 2^n, \text{ αν } n > 1.$$

Αποδείξτε επαγωγικά ότι για κάθε ακέραιο $n \geq 3$, $f(n) \leq 3n * 2^{n-2}$.

5. Βρείτε την τάξη (βάσει των συμβολισμών O , Ω και Θ) της χρονικής πολυπλοκότητας $T(n)$ των ακόλουθων αλγορίθμων:

```
a. Procedure Mystery(integer n) {  
    for (i=1; i <= n; i++) {  
        for (j = i+1; j <= n; j++)  
            x = x+1;  
    }  
}
```

```
b. Procedure Puzzle(integer n) {  
    for (i=1; i <= n; i++)  
        for (j=0; j < 10; j++)  
            for (k=n; k <= n+5; k++)  
                x = x+1;  
}
```

```
c. Procedure Foo(integer n) {  
    for (i = 1; i <= n; i++) {  
        x = n;  
        while (x > 0)  
            x = x - i;  
    }  
}
```

Σημείωση: Το $H_n = 1 + 1/2 + 1/3 + \dots + 1/n$ ονομάζεται αρμονικός αριθμός και είναι γνωστό ότι $H_n = O(\ln n)$.