

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**ΕΠΛ 035: Δομές Δεδομένων και Αλγόριθμοι για Ηλεκτρολόγους Μηχανικούς και
Μηχανικούς Υπολογιστών**

Χειμερινό Εξάμηνο 2012

**ΑΣΚΗΣΗ 3
Ανάλυση Πολυπλοκότητας**

Διδάσκων Καθηγητής: Παναγιώτης Ανδρέου

**Ημερομηνία Υποβολής: 08/10/2012
Ημερομηνία Παράδοσης: 18/10/2012**

Υπεύθυνος Εργαστηρίου: Πέτρος Παναγή (petrosp@cs.ucy.ac.cy)

ΠΕΡΙΓΡΑΦΗ

Σε αυτή την άσκηση καλείστε να αναλύσετε και να υπολογίσετε το χρόνο εκτέλεσης διαφόρων λειτουργιών χρησιμοποιώντας τους ορισμούς που διδαχτήκατε, την μέθοδο της μαθηματικής επαγωγής και την μέθοδο της αντικατάστασης.

Άσκηση 1 (15 μονάδες)

Χρησιμοποιείτε τον ορισμό « $f(x)$ είναι $O(g(x))$ » για να δείξετε ότι:

α) $x^4 + 9x^4 + 4x + 7$ είναι $O(x^4)$

β) $\frac{(x^2+1)}{(x+1)}$ είναι $O(x)$

γ) $x^2 + 4x + 17$ είναι $O(x^3)$,

αλλά επίσης, ότι το x^3 δεν είναι $O(x^2 + 4x + 17)$

Άσκηση 2 (15 μονάδες)

Χρησιμοποιείτε την μαθηματική επαγωγή για να αποδείξετε ότι το άθροισμα των πρώτων n θετικών περιττών αριθμών είναι ίσο με n^2 .

Άσκηση 3 (15 μονάδες)

Υποθέστε ότι κάποιος έχει αναλύσει τον χρόνο εκτέλεσης μίας σειράς αλγορίθμων και δημιούργησε μία λίστα με τις πολυπλοκότητες των συναρτήσεων τους. Καλείστε να κατατάξατε τις συναρτήσεις αυτές σε κατηγορίες έτσι ώστε δύο συναρτήσεις $f(n)$ και $g(n)$ να ανήκουν στην ίδια

κατηγορία αν και μόνο αν ανήκουν στην ίδια τάξη (δηλαδή αν $f(n) \in \Theta(g(n))$ ή $g(n) \in \Theta(f(n))$).

Στην συνέχεια καλείστε να ταξινομήσετε τις κατηγορίες σε αύξουσα σειρά ως προς την τάξη τους.

\sqrt{n}	n	3^n	$n^2 + \log(n)$
6	$\log(n)$	$n - n^3 + 7n^5$	$n \log(n)$
$n!$	$n^2 + 5n^3$	n^2	n^3

Άσκηση 4 (40 μονάδες)

Να αναλύσετε τον χρόνο εκτέλεσης χειρίστης περίπτωσης των πιο κάτω σαν συνάρτηση του n . Υποθέστε, χωρίς βλάβη της γενικότητας, ότι το n είναι δύναμη του 2.

A)

```
sum=0;
for ( i=1; i<=n; i*=2 )
    for ( j=1; j<=i; j++ )
        sum++;
```

B)

```
r=0;
for ( i=1; i<=sqrt(n); i++ )
    for ( j=1; j<=n; j*=2 )
        if ( n % 2 == 0 ) //το n είναι άρτιος
            for ( k=1; k<=n; k++ )
                r++;
        else //το n είναι περιττός
            r--;
```

Άσκηση 5 (15 μονάδες)

Να υπολογίσετε το χρόνο εκτέλεσης του παρακάτω αναδρομικού προγράμματος λύνοντας οποιοσδήποτε αναδρομικές εξισώσεις συναντήσετε με την μέθοδο της αντικατάστασης. Υποθέστε χωρίς βλάβη της γενικότητας ότι το n είναι δύναμη του 2.

```
SplitSum( int n ) {
    int sum = 0;
    for ( j=1; j<=n; j++ ) sum++;
    if ( n > 1 ) return SplitSum( n/2 );
    else return 1;
}
```

ΟΔΗΓΙΕΣ

- Οι άσκηση σας θα πρέπει να παραδοθεί σε έντυπη και ηλεκτρονική μορφή μέσω email στον υπεύθυνο βοηθό.
- Το όνομα του αρχείου θα ονομάζεται με τον εξής τρόπο:
epi035.ex<αρ. άσκησης>.<ταυτότητα>.<ext> (π.χ., epi035.ex3.123456.doc)
- Στην πρώτη σελίδα του αρχείου θα πρέπει να αναγράφεται το όνομα και η ταυτότητά σας.
- Μη τήρηση των ημερομηνιών παράδοσης των εργασιών συνεπάγεται τις ανάλογες βαθμολογικές επιπτώσεις (μέχρι τον μηδενισμό της εργασίας).
- Οι προγραμματιστικές ασκήσεις θα ελέγχονται από ειδικό πρόγραμμα για την ανίχνευση των αντιγραφών. Οι αντιγραμμένες εργασίες θα μηδενίζονται και για τους αντιγραφείς θα εφαρμόζονται οι κανόνες τού Πανεπιστημίου. Αποφύγετε λοιπόν την αντιγραφή προγραμμάτων από άλλους συναδέλφους σας, διότι έτσι εκτίθετε και αυτούς και τον εαυτό σας στον κίνδυνο μηδενισμού και πειθαρχικής δίωξης.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ!!!