

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΛ 231: Δομές Δεδομένων και Αλγόριθμοι

ΘΕΩΡΗΤΙΚΗ ΑΣΚΗΣΗ 1

Στόχος: 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

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

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

Άσκηση 3

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

Να αναλύσετε τον χρόνο εκτέλεσης χειριστής 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<=√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

Να υπολογίσετε το χρόνο εκτέλεσης του παρακάτω αναδρομικού προγράμματος λύνοντας οποιεσδήποτε 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;
}
```