Αλγόριθμοι και Πολυπλοκότητα: 1ο Σύνολο Ασκήσεων

1. Έστω $f, f_1, f_2, g, g_1, g_2, h$ θετικές συναρτήσεις. Αποφασίστε αν οι παρακάτω προτάσεις είναι αληθείς ή ψευδείς. Δικαιολογήστε τις απαντήσεις σας.

1. Αν $f(n) = o(g(n))$ τότε $f(n) = O(g(n))$
2. Αν $f(n) = O(g(n))$ και $f(n) = \Omega(h(n))$ τότε $g(n) = \Theta(h(n))$
3. Αν $f_1(n) = O(g_1(n))$ και $f_2(n) = O(g_2(n))$ τότε

   $i.$ $f_1(n) + f_2(n) = O(max\{g_1(n), g_2(n)\})$
   $ii.$ $f_1(n) \cdot f_2(n) = O(g_1(n) \cdot g_2(n))$.
4. $4n^2 + 5n - 9 = \Omega(10n^2)$.
5. $\log(n!) = \Theta(n \log n)$.
6. $f(n) + g(n) = \Theta(\min(g(n), f(n)))$
7. $n + 2\sqrt{n} \neq \Omega(n\sqrt{n})$
8. Έστω $P(n) = a_k n^k + a_{k-1} n^{k-1} + \cdots + a_1 n + a_0$ πολυώνυμο k βαθμού, $a_k > 0$. Να δειχθεί ότι
$$P(n) = \Theta(n^k).$$

2. Δίνονται οι παρακάτω συναρτήσεις. Χωρίστε τη λίστα των συναρτήσεων σε κλάσεις, τέτοιες ώστε οι $f(n)$ και $g(n)$ να ανήκουν στην ίδια κλάση αν και μόνο αν $f(n) = \Theta(g(n))$. Στη συνέχεια επιλέξτε έναν εκπρόσωπο από κάθε κλάση και κατατάξτε τους κατά αύξουσα σειρά πολυπλοκότητας.

$$\begin{array}{ccccc}
\binom{n}{2} & n\log n & \sum_{k=1}^{n}\frac{1}{k} & 8n^2 & \log\sqrt{\log n} \\
n! & \log\log n & n^{\log n} & \log n! & 4^{\log n} \\
\sum_{k=0}^{n}\binom{n}{k} & 2^{\log^2 n} & 10^{100} & 2^n & \log n \\
(\sqrt{2})^{\log n} & (n-1)! & \log n^n & 5^{800} & 5\sqrt{n}
\end{array}$$

3. Για κάθε ζεύγος εκφράσεων (A, B) του παρακάτω πίνακα αποφασίστε αν το A είναι O, o, Ω, ω ή Θ του B. Υποθέστε ότι οι $k \geq 1, \epsilon > 0$ και $c > 1$ είναι σταθερές. Απαντήστε σημειώνοντας ένα 'ναι' ή ένα 'όχι' σε κάθε θέση του πίνακα.

	A	B	O	o	Ω	ω	Θ
a.	$\log^k n$	n^ϵ					
b.	n^k	c^n					
c.	2^n	$2^{n/2}$					
d.	$n^{\log c}$	$c^{\log n}$					
στ.	$n!$	n^n					

4. Να βρεθεί η πολυπλοκότητα των ακόλουθων αλγορίθμων.

Αλγόριθμος 1

A

```
sum = 0
for i = 1 to 4n do
    for j = 1 to 2n^2 with step 2 do
        for k = n to n/2 with step -1 do
            sum = sum + 1
        end for
    end for
end for
```

Αλγόριθμος 2

β

```
sum = 0
for i = 1 to n - 1 do
    for j = 1 to n * i do
        for k = 1 to j do
            sum = sum + 1
        end for
    end for
end for
```

5.

1. Να βρεθεί η βέλτιστη, η κατά μέσο όρο και η χείριστη πολυπλοκότητα της Σειριακής Αναζήτησης. Υποθέστε ότι:

   i. Η ακολουθία προς ταξινόμηση αποτελείται από διαφορετικά στοιχεία.

   ii. Η πιθανότητα ένα στοιχείο να μην υπάρχει στον πίνακα $A[1..n]$ είναι p.

2. Αν ο πίνακας A έχει μόνο m διαφορετικά στοιχεία τότε να δειχθεί ότι η κατά μέσο όρο πολυπλοκότητα της Σειριακής Αναζήτησης είναι $\Theta(m)$ για $n \to \infty$. Από αυτό το αποτέλεσμα προκύπτει ότι αν $m =$ σταθερό, τότε η πολυπλοκότητα της Σειριακής Αναζήτησης είναι $\Theta(1)$, δηλαδή σταθερής τάξης.

6.

1. Να βρεθεί η κατά μέσο όρο πολυπλοκότητα της Ενθετικής Ταξινόμησης (Insertion Sort).

2. Να βρεθεί το πλήθος των συγκρίσεων κατά μέσο όρο της Ενθετικής Ταξινόμησης με Δυαδική Αναζήτηση.

3. Να βρεθεί η κατά μέσο όρο πολυπλοκότητα της Ενθετικής Ταξινόμησης με Δυαδική Αναζήτηση.

4. Σχολιάστε τα συμπεράσματά σας. Ποιά εκδοχή της Ενθετικής Ταξινόμησης είναι καλύτερη και γιατί;

7. Να μελετηθεί η περίπτωση υποδιαίρεσης του πίνακα $A[1..n]$ σε περισσότερα από δυο τμήματα στην Merge-Sort. Πιο συγκεκριμένα

1. Να δοθεί αναδρομικός αλγόριθμος, ο οποίος να διαιρεί τον A σε τρία τμήματα. Υποθέστε $n = 3^k$.

2. Να βρεθεί η πολυπλοκότητα του αλγορίθμου του (1). Σχολιάστε τα αποτελέσματά σας σε σχέση με την διαίρεση του A σε δύο τμήματα.

3. Να αποδειχθεί η ορθότητα του αλγορίθμου του (1) με την χρήση της επαγωγής και της αναλλοίωτης συνθήκης.