

Ασκήσεις Επανάληψης

Επιμέλεια: Αλέξανδρος Αργυρόπουλος

Ενότητα 1 : Αριθμητική υπολογιστή και στοιχεία Γραμμικής Άλγεβρας

1. Να μετατραπεί ο αριθμός $1011000_{(2)}$ σε δεκαεξαδικό και σε οκταδικό σύστημα .

2. Έστω ο πίνακας $A = \begin{bmatrix} \frac{1}{2} & \frac{1}{12} & \frac{3}{2} \\ -3 & 0 & -1 \\ 2 & \frac{1}{3} & 24 \end{bmatrix}$. Να μετατρέψετε τον αριθμό $\det(A)_{(10)}$ σε δυαδικό σύστημα .

3. Έστω πίνακες $A \in \mathbb{R}^{20 \times 3}$, $B^T \in \mathbb{R}^{2026 \times 2^{20}}$. Ποια είναι η καλύτερη χρήση της προσεταιριστικής ιδιότητας για το γινόμενο : $A \cdot A^T \cdot B$.

4. Έστω αριθμητική κινητής υποδιαστολής $\mathcal{M}(b, t, m, M)$, να δώσετε ερμηνεία για τους αριθμούς b, t, m, M . Για την $\mathcal{M}(10, 4, -4, 4)$ να υπολογιστούν τα $fl(x_1 + x_2)$, $fl(x_1 \cdot x_2)$ για $x_1 = 0.1253 * 10^3$, $x_2 = 0.8590 * 10^2$.

5. Να δοθεί αλγόριθμος σε ψευδογλώσσα για τη συνάρτηση **MatrP(A,B)** του Θέματος 3, Ε1α . Ποια η πολυπλοκότητά του ; Να εκτιμηθεί , επίσης , η πολυπλοκότητα της συνάρτησης **Plist(A, n)**

Ενότητα 2 : Ανάπτυξη και κατανόηση κώδικα σε L^AT_EX

1. Να γράψετε τι εμφανίζει ο εξής κώδικας :

$$\int_{\mu}^{\nu} \ln(yx^{35}) \, dy \in \mathbb{N}$$

2. Να γράψετε (έχοντας δεδομένα τα απαραίτητα πακέτα) τον κώδικα που θα εμφανίσει την εξής παράσταση :

$$\frac{f}{g}(x) + h^{3\alpha}(x) = \lim_{N \rightarrow +\infty} \left[\sum_{k=0}^N \binom{x}{k} \right] \cdot \pi x$$

Ενότητα 3 : Προγραμματισμός σε Python

1. α) Έστω ότι έχετε υλοποιήσει σε πρόγραμμα μία συνάρτηση **MatP(A, B)** η οποία υπολογίζει το γινόμενο των πινάκων A, B . Να φτιάξετε συνάρτηση **Plist(A, n)** που δέχεται τετραγωνικό πίνακα A και επιστρέφει σε λίστα τους πίνακες $A, A^2, \dots, A^n$ ($n \in \mathbb{N}$) .
β) Να τροποποιήσετε την συνάρτηση ώστε να λαμβάνει υπόψη την περίπτωση ο A να είναι μηδενοδύναμος , αποφεύγοντας περιττούς υπολογισμούς . (!) Θεωρούμε πως γίνεται ο έλεγχος $A = / \neq a$ που εξετάζει αν όλα τα στοιχεία του A είναι ίσα (/άνισα) με a .

2. α) Να φτιαχτεί συνάρτηση `IsIntlist(l)` η οποία δέχεται μία λίστα `l` και επιστρέφει κατάλληλη τιμή τύπου `Bool`, αναφορικά με το αν η `l` περιέχει μόνο ακέραιους .
- β) Να αποφανθείτε τι πρόκειται να εκτυπώσει ο παρακάτω κώδικας , θεωρώντας υλοποιημένη την συνάρτηση του ερωτήματος 2α) . Να αιτιολογήσετε την απάντησή σας .

<code>n = int (input("Πλήθος τιμών ; "))</code>	Είσοδος : <code>n = 3</code>
<code>l = []</code>	<code>x = 1</code>
<code>for i in range(n):</code>	<code>x = 2</code>
<code>x = input(f'Δώσε τον {i+1}ο αριθμό: ")</code>	<code>x = 3</code>
<code>l.append(x)</code>	
<code>print(IsIntlist(l))</code>	

3. (!) Σε όλες τις παρακάτω διεργασίες η λίστα του ορίσματος να παραμένει αναλλοίωτη .
- α) Αποφεύγοντας τη χρήση της συνάρτησης `min` , να φτιάξετε μία δική σας συνάρτηση `lmin(l)` , που επιστρέφει το ελάχιστο στοιχείο της `l` .

β) Να υλοποιήσετε μία συνάρτηση `merge(l)` , η οποία επιστρέφει αντίγραφο της `l` στο οποίο το ελάχιστο στοιχείο έχει μεταφερθεί στην αρχή της .

Υπόδειξη : Μπορείτε να χρησιμοποιήσετε τη μέθοδο `remove` των λιστών . (π.χ.) `[1,3,4,3].remove(3) → [1,4,3]` .

γ) Να γράψετε μία συνάρτηση `my_sort(l)` , η οποία θα επιστρέφει διατεταγμένο αντίγραφο της `l` , σε αύξουσα σειρά , χρησιμοποιώντας τις προηγούμενες συναρτήσεις .

δ) Τέλος να εμπλουτίσετε "το πρόγραμμα" με μία συνάρτηση `main()` , η οποία θα ζητάει από το χρήστη το πλήθος και τις τιμές των στοιχείων της `l` , θα εκτυπώνει την `l` , την `l` σε αύξουσα σειρά , το ελάχιστο και το μέγιστο στοιχείο της , εφόσον είναι λίστα ακεραίων . Σε αντίθετη περίπτωση θα εκτυπώνει την `l` και μήνυμα λάθους .