

# ΠΛΗ10 – ΤΕΣΤ 31

## Θέμα 2: Ψευδογλώσσα και Αλγόριθμοι

### Ερώτημα 1

Έστω μονοδιάστατος πίνακας  $A$ ,  $N$  ακεραίων αριθμών. Σχεδιάστε διάγραμμα ροής προγράμματος (ΔΡΠ) για την αναζήτηση στον πίνακα  $A$  και εν συνεχεία την εκτύπωση της τιμής και της θέσης του πλησιέστερου στοιχείου σε δεδομένο αριθμό  $NUM$ . Ο πίνακας  $A$ , το πλήθος των στοιχείων  $N$  και ο αριθμός  $NUM$  θεωρούνται δεδομένα και δεν χρειάζεται να διαβαστούν.

## **Θέμα 3: Ερωτήματα σε Γλώσσα C**

### Ερώτημα 1

Τι θα εκτυπωθεί μετά την εκτέλεση του παρακάτω προγράμματος; Αιτιολογήστε την απάντησή σας:

```
#include <stdio.h>
```

```
int seira(int n);
```

```
int main()  
{  
    printf("%d\n", seira(5));  
}
```

```
int seira(int n)  
{  
    if (n == 0)  
        return 1;  
    if (n%2 == 0)  
        return seira(n-2) + seira(n-1);  
    else  
        return seira(n-1)+3;  
}
```

## Ερώτημα 2

Γράψτε στη γλώσσα προγραμματισμού C τη συνάρτηση `int_n_to_10 (int n, int num)`, η οποία μετατρέπει έναν θετικό τριψήφιο αριθμό του  $n$ -δικού συστήματος αρίθμησης σε δεκαδικό αριθμό τον οποίο και επιστρέφει. Η συνάρτηση δέχεται ως παραμέτρους τη βάση του αριθμητικού συστήματος  $n$  (η τιμή του  $n$  δεν είναι μεγαλύτερη του 10) και τον θετικό αριθμό του  $n$ -δικού συστήματος αρίθμησης  $num$ . Δεν απαιτείται αμυντικός προγραμματισμός για τις τιμές του  $n$  και του  $num$ .

Ερώτημα 3

Υλοποιήστε πρόγραμμα στη γλώσσα προγραμματισμού C, το οποίο θα διαβάζει τον πίνακα ακεραίων  $\text{pin}$ , διαστάσεων  $5 \times 7$  και στη συνέχεια θα βρίσκει το ελάχιστο στοιχείο κάθε στήλης, καθώς και τη γραμμή στην οποία βρέθηκε. Τα αποτελέσματα θα αποθηκεύονται στον πίνακα  $\text{min}$ , διαστάσεων  $2 \times 7$ , στην 1<sup>η</sup> γραμμή του οποίου θα αποθηκεύεται η τιμή του ελάχιστου στοιχείου κάθε στήλης και στη 2<sup>η</sup> γραμμή η θέση (γραμμή του αρχικού πίνακα) στην οποία βρέθηκε το στοιχείο αυτό. Σε περίπτωση που το ελάχιστο στοιχείο κάποιας στήλης συναντάται σε περισσότερες θέσεις της, να αποθηκεύεται η θέση αυτού που συναντάται πρώτο κατά τη σάρωση της στήλης, δηλαδή αυτό με το μικρότερο δείκτη γραμμής. Τέλος το πρόγραμμα θα πρέπει να εμφανίζει τα αποτελέσματα σε τρεις στήλες στην οθόνη, ως εξής: στην 1<sup>η</sup> στήλη θα εμφανίζεται ο αριθμός της εκαστοτε στήλης του πίνακα  $\text{pin}$ , στη 2<sup>η</sup> στήλη θα εμφανίζεται ο αριθμός της γραμμής του πίνακα  $\text{pin}$  στην οποία βρίσκεται το ελάχιστο στοιχείο της αντίστοιχης στήλης και στην 3<sup>η</sup> στήλη θα εμφανίζεται η ελάχιστη τιμή της αντίστοιχης στήλης του πίνακα  $\text{pin}$ . Για την εμφάνιση των αποτελεσμάτων πρέπει να γίνει αποκλειστικά χρήση του πίνακα  $\text{min}$ .

#### Ερώτημα 4

Να γραφεί στη γλώσσα προγραμματισμού C η συνάρτηση:

```
void average(int array[][N], int N)
```

η οποία δέχεται ως όρισμα έναν πίνακα ακεραίων δύο διαστάσεων ( $N \times N$ ) και για κάθε γραμμή του πίνακα τυπώνει στην οθόνη το μέσο όρο των θετικών (δηλαδή μεγαλύτερων του 0) στοιχείων. Στην περίπτωση που δεν υπάρχουν θετικά στοιχεία σε κάποια γραμμή, να τυπώνει μήνυμα αδυναμίας πραγματοποίησης του υπολογισμού.

## **Θέμα 4: Ερωτήματα σε Δομές Δεδομένων**

### **Ερώτημα 1**

Δίνονται οι τιμές: 15, 5, 40, 1, 30, 2, 44, 25.

1. Σχηματίστε το πλήρες δυαδικό δένδρο αναζήτησης οι κόμβοι του οποίου έχουν τις τιμές αυτές.
2. Δώστε την προδιατεταγμένη και την ενδοδιατεταγμένη διαπέραση του δένδρου
3. Θεωρήστε ότι το δένδρο σαρώνεται με τη μεταδιατεταγμένη διαπέραση και τα δεδομένα των κόμβων του ωθούνται σε μία στοίβα με διαδοχικές λειτουργίες push. Στην συνέχεια όλα τα δεδομένα απωθούνται από τη στοίβα με διαδοχικές λειτουργίες pop και εισάγονται σε ένα αρχικά κενό δένδρο-σωρό. Σε ποιο επίπεδο (βάθος) θα βρεθεί τελικά η τιμή 40, αν θεωρήσουμε ότι το βάθος της ρίζας του δένδρου είναι 0;

## Ερώτημα 2

Έστω ότι ένα δυαδικό δένδρο αναζήτησης ορίζεται ως εξής:

```
typedef struct bin_tree_node
{
    int val;
    struct bin_tree_node *left, *right;
} *bin_tree_ptr;
```

1. Υλοποιήστε μια συνάρτηση `swap_subtrees(bin_tree b)` που εναλλάσσει τα δύο υποδένδρα του κόμβου στον οποίο δείχνει ο δείκτης `b`.
2. Υλοποιήστε αναδρομική συνάρτηση `recursive_swap_subtrees(bin_tree_ptr b)` που, όπως και η `swap_subtrees(bin_tree_ptr b)` εναλλάσσει τα δύο υποδένδρα του κόμβου στον οποίο δείχνει ο δείκτης `b`, αλλά και αναδρομικά επιτελεί την ίδια λειτουργία σε κάθε κόμβο των δύο υποδένδρων του κόμβου αυτού.
3. Αν ο δείκτης `b` (τύπου `bin_tree_ptr`) δείχνει στη ρίζα ενός δυαδικού δένδρου αναζήτησης (του οποίου η ενδοδιατεταγμένη διαπέραση δίνει τα στοιχεία σε αύξουσα σειρά) ποια θα είναι η ενδοδιατεταγμένη διαπέραση του δένδρου μετά την κλήση της συνάρτησης `recursive_swap_subtrees(b)` σε σχέση με την ενδοδιατεταγμένη διαπέραση του δένδρου πριν την κλήση της;