

Αλγόριθμοι και Πολυπλοκότητα
Άσκηση 1η
Παράδοση: 24 Απριλίου 2020

Ερώτημα 1.

- (α) Διατάξτε τις ακόλουθες δέκα συναρτήσεις με βάση τον αυξητικό τους χαρακτήρα, δηλαδή εάν η συνάρτηση $f(n)$ είναι αμέσως πριν τη συνάρτηση $g(n)$ στη σειρά, θα πρέπει να ισχύει $f(n) = O(g(n))$. Σημειώνεται ότι όλοι οι λογάριθμοι (\log) έχουν βάση το 2.

$$2n + n \log n, \quad 3 \log n, \quad \left(\sum_{i=1}^n i \right)^2, \quad \frac{4^n}{n^4}, \quad n^{7/8},$$
$$2^n, \quad 10^{\log n}, \quad n \log n!, \quad \sqrt{\log 5n}, \quad n^{\log n}.$$

- (β) Να επιλύσετε τις ακόλουθες αναδρομικές σχέσεις. Δίνεται ότι για όλες τις σχέσεις ισχύει $T(1) = O(1)$.

$$\begin{aligned} T(n) &= 8T(n/2) + n^2 \\ T(n) &= 3T(n/3) + n \\ T(n) &= T(4n/5) + 1. \end{aligned}$$

Δεν απαιτούνται εξηγήσεις για τις απαντήσεις σας στο Ερώτημα 1.

Ερώτημα 2.

Επιλύστε την παρακάτω αναδρομική σχέση με τη μέθοδο δέντρου αναδρομής ή με τη μέθοδο εικασίας και επαγωγικής απόδειξης.

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 5T(n/5) + n^2 \quad \text{για } n > 1. \end{aligned}$$

Ερώτημα 3.

- (α) Έστω ένα κατευθυνόμενο γράφημα G . Εκτελούμε αναζήτηση σε βάθος (DFS) στο G και για μία ακμή του (u, v) δίνεται ότι ισχύει $\text{pre}[u] < \text{pre}[v]$. Τι τύπος ακμής μπορεί να είναι η (u, v) ; Επιλέξτε έναν ή περισσότερους από τους ακόλουθους τύπους: ακμή δέντρου, ακμή μπροστά, ακμή πίσω, ακμή διασταύρωσης.
- (β) Τροποποιήστε τον ψευδοκώδικα του DFS έτσι ώστε να υπολογίζει το μέγιστο ύψος δέντρου που προκύπτει στο δάσος DFS.
- (γ) Έστω ένα μη κατευθυνόμενο γράφημα G και έστω e μία ακμή του. Δώστε έναν αλγόριθμο γραμμικού χρόνου που να αποφασίζει αν υπάρχει ή όχι κύκλος στο G που να περιέχει την ακμή e .

Ερώτημα 4.

Έστω n αθλητές λαμβάνουν μέρος σε ένα πρωτάθλημα όπου αγωνίζονται ένας εναντίον ενός. Υποθέστε ότι το n είναι δύναμη του 2. Το πρωτάθλημα αποτελείται από μία σειρά αγωνιστικών. Σε κάθε αγωνιστική όλοι οι αθλητές λαμβάνουν μέρος σε κάποιον αγώνα, γίνονται δηλαδή $n/2$ αγώνες. Επιπλέον κάθε αθλητής πρέπει να αγωνιστεί στη διάρκεια του πρωταθλήματος ακριβώς μία φορά με κάθε άλλο αθλητή. Δώστε έναν αλγόριθμο *διαίρει-και-βασίλευε* ο οποίος να υπολογίζει ένα πρόγραμμα αγώνων για το πρωτάθλημα που να ικανοποιεί τους παραπάνω περιορισμούς. Αναλύστε τον χρόνο εκτέλεσης του αλγορίθμου σας. Για παράδειγμα, αν $n = 4$ ο αλγόριθμος μπορεί να δώσει ως λύση το εξής πρόγραμμα αγώνων:

Αγώνες 1ης αγωνιστικής: αθλητής 1 με αθλητή 2, αθλητής 3 με αθλητή 4.

Αγώνες 2ης αγωνιστικής: αθλητής 1 με αθλητή 3, αθλητής 2 με αθλητή 4.

Αγώνες 3ης αγωνιστικής: αθλητής 1 με αθλητή 4, αθλητής 2 με αθλητή 3.