Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Αλγόριθμοι και Πολυπλοκότητα
Διδάσκοντες: Δημήτρης Φωτάκης, Δώρα Σούλιου
1η Σειρά Γραπτών Ασκήσεων - Ημ/νία Παράδοσης 27/10/2016

Άσκηση 1: Ασυμπτωτικός Συμβολισμός, Αναδρομικές Σχέσεις.

(α) Να ταξινομήσετε τις παρακάτω συναρτήσεις σε αύξουσα σειρά τάξης μεγέθους, να βρείτε δηλαδή μια διάταξη $g_1, g_2, g_3, \ldots$ τέτοια ώστε $g_1 = \mathrm{O}(g_2)$, $g_2 = \mathrm{O}(g_3)$, κοκ. Σε αυτή τη διάταξη, να επισημάνετε τις συναρτήσεις που έχουν ίδια τάξη μεγέθους.

$n^{3/2}$	$3^{(\log_2 n)^3}$	$\log(n!)/(\log n)^3$	$n2^{2^{2^{100}}}$
$\log(\binom{n}{\log n})$	$(\log n)^2/\log\log n$	$\log^4 n$	$(\sqrt{n})!$
$\binom{n}{6}$	$n^3/(\log n)^8$	$(\log_2 n)^{\log_2 n}$	$\log\left(\binom{2n}{n}\right)$
$n\sum_{k=0}^{n}\binom{n}{k}$	$n^{\log_2(n!)}$	$\sum_{k=1}^{n} k2^k$	$\sum_{k=1}^{n} k2^{-k}$

(β) Να υπολογίσετε την τάξη μεγέθους $\Theta()$ των λύσεων των παρακάτω αναδρομικών σχέσεων. Για όλες τις σχέσεις, να θεωρήσετε ότι $T(1) = \Theta(1)$.

1. $T(n) = 2T(n/3) + n\log n$
2. $T(n) = 3T(n/3) + n\log n$
3. $T(n) = 4T(n/3) + n\log n$
4. $T(n) = T(n/2) + T(n/3) + n$
5. $T(n) = T(n/2) + T(n/3) + T(n/6) + n$
6. $T(n) = T(n-1) + \log n$
7. $T(n) = T(n^{5/6}) + \Theta(\log n)$
8. $T(n) = T(n/4) + \sqrt{n}$.

Άσκηση 2: Ταξινόμηση

(α) Θεωρούμε έναν πίνακα $A[1\ldots n]$ με n στοιχεία και τους υποπίνακες $A_1[1\ldots \frac{n}{k}]$, $A_2[\frac{n}{k}+1\ldots 2\frac{n}{k}]$, $\ldots, A_k[(k-1)\frac{n}{k}+1\ldots n]$ που προκύπτουν από την διαμέριση του A σε k τμήματα με n/k στοιχεία το καθένα (για απλότητα, υποθέτουμε ότι το n είναι πολλαπλάσιο του k, μπορείτε ακόμη να υποθέσετε ότι τα n και k είναι δυνάμεις του 2). Θα λέμε ότι ο πίνακας A είναι *ταξινομημένος κατά* k *μέρη* όταν για κάθε i και j, με $1 \le i < j \le k$, κάθε στοιχείο του υποπίνακα A_i είναι μικρότερο ή ίσο από κάθε στοιχείο του υποπίνακα A_j. Δηλ. τα στοιχεία του A έχουν ταξινομηθεί "εξωτερικά", μεταξύ των υποπινάκων, αλλά όχι κατ' ανάγκη και "εσωτερικά" στον ίδιο υποπίνακα. Π.χ., ο πίνακας $A = \big[[12, 14, 20, 18], [25, 22, 29, 32], [37, 42, 34, 50], [67, 59, 52, 76]\big]$ είναι ταξινομημένος κατά 4 μέρη.

1. Να διατυπώσετε συγκριτικό αλγόριθμο με χρόνο εκτέλεσης $O(n\log k)$ που ταξινομεί κατά k μέρη έναν πίνακα A με n στοιχεία. Να δείξετε ακόμη ότι ο αλγόριθμός σας είναι βέλτιστος, δηλ. ότι κάθε συγκριτικός αλγόριθμος ταξινόμησης κατά k μέρη ενός πίνακα με n στοιχεία έχει χρόνο εκτέλεσης χειρότερης περίπτωσης $\Omega(n\log k)$.

2. Να διατυπώσετε συγκριτικό αλγόριθμο με χρόνο εκτέλεσης $O(n \log \frac{n}{k})$ που ταξινομεί πλήρως έναν πίνακα A με n στοιχεία που αρχικά είναι ταξινομημένος κατά k μέρη. Να εξηγήσετε γιατί κάθε συγκριτικός αλγόριθμος για αυτό το πρόβλημα πρέπει να έχει χρόνο εκτέλεσης $\Omega(n \log \frac{n}{k})$.

(β) Πριν την επίδειξη γραπτών στο μάθημα "Προγραμματισμός Η/Υ", ταξινομούμε τα γραπτά σε αλφαβητική σειρά. Πέρυσι, κατά τη μεταφορά των γραπτών και μόλις $30'$ πριν την επίδειξή τους, η στοίβα σκορπίστηκε στο πάτωμα. Όταν μαζέψαμε τα γραπτά και προσπαθώντας να τα ταξινομήσουμε πάλι, παρατηρήσαμε ότι κάθε γραπτό βρισκόταν σε απόσταση το πολύ k θέσεων (προς τα πάνω ή προς τα κάτω) από την αρχική του θέση στην ταξινομημένη στοίβα. Έπρεπε λοιπόν να ταξινομήσουμε πολύ γρήγορα μια στοίβα με n γραπτά, που ήταν όμως *k-προταξινομημένη*, με την έννοια ότι κάθε γραπτό βρισκόταν σε απόσταση το πολύ k θέσεων από την τελική του θέση στην ταξινομημένη στοίβα.

1. Να διατυπώσετε αποδοτικό συγκριτικό αλγόριθμο για την ταξινόμηση ενός k-προταξινομημένου πίνακα με n στοιχεία. Να αιτιολογήσετε συνοπτικά την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας.
2. Να δείξετε ότι κάθε συγκριτικός αλγόριθμος για την ταξινόμηση ενός k-προταξινομημένου πίνακα με n στοιχεία έχει χρόνο εκτέλεσης χειρότερης περίπτωσης $\Omega(n \log k)$.

Άσκηση 3: Αναζήτηση

(α) Κάποια άλλη χρονιά, μόλις $5'$ πριν την επίδειξη γραπτών στο μάθημα "Προγραμματισμός Η/Υ", αφού είχαμε ταξινομήσει τα γραπτά σε αλφαβητική σειρά, κάποιος "έκοψε" τη στοίβα σε μια θέση k και τοποθέτησε τα πρώτα k γραπτά στο τέλος. Το αποτέλεσμα ήταν μια *περιστραμμένη ταξινομημένη* στοίβα γραπτών $a_{k+1}, a_{k+2}, \dots, a_{n-1}, a_n, a_1, a_2, \dots, a_{k-1}, a_k$, όπου $a_1 \leq a_2 \leq \cdots \leq a_{k-1} \leq a_k \leq a_{k+1} \leq \cdots \leq a_{n-1} \leq a_n$. Για να ταξινομήσουμε πάλι πλήρως τη στοίβα, πρέπει να βρούμε το ελάχιστο στοιχείο της περιστραμμένης ταξινομημένης στοίβας (και την πρώτη θέση όπου αυτό εμφανίζεται), ώστε να επαναφέρουμε τα k γραπτά από το τέλος στην αρχή της στοίβας. Να διατυπώσετε έναν αποδοτικό συγκριτικό αλγόριθμο που βρίσκει το ελάχιστο στοιχείο (και την πρώτη θέση όπου αυτό εμφανίζεται) ενός περιστραμμένου ταξινομημένου πίνακα $A[1 \dots n]$ με n στοιχεία. Να αιτιολογήσετε συνοπτικά την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας.

(β) Πρόκειται να πάμε ένα μεγάλο ταξίδι που θα διαρκέσει $k \geq 2$ ημέρες. Έχουμε επιλέξει τους $n > k$ σταθμούς του ταξιδιού (και τη σειρά με την οποία θα τους επισκεφθούμε) και έχουμε υπολογίσει τις αποστάσεις $d_1, d_2, \dots, d_n$ μεταξύ τους (d_1 είναι η απόσταση του πρώτου σταθμού από την αφετηρία, d_2 είναι η απόσταση του πρώτου από τον δεύτερο σταθμό, κοκ., όλες οι αποστάσεις είναι θετικοί ακέραιοι). Για να μην κουραστούμε, θέλουμε να προγραμματίσουμε το ταξίδι ώστε να ελαχιστοποιήσουμε τη μέγιστη απόσταση που θα διανύσουμε μέσα σε μία ημέρα (μπορούμε να σταματήσουμε για διανυκτέρευση μόνο σε κάποιον από τους επιλεγμένους σταθμούς). Να διατυπώσετε έναν αποδοτικό αλγόριθμο για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας.

Άσκηση 4: Εντοπισμός Επαναλαμβανόμενου Στοιχείου

Θεωρούμε έναν πίνακα $A[1 \dots n]$ με στοιχεία φυσικούς αριθμούς στο σύνολο $[n] = \{1, \dots, n\}$, όπου ένας αριθμός $k \in [n]$ εμφανίζεται τουλάχιστον δύο φορές και κάθε άλλος αριθμός στο σύνολο $[n] \setminus \{k\}$ εμφανίζεται το πολύ μία φορά στον πίνακα A. Να διατυπώσετε αλγόριθμο γραμμικού χρόνου που υπολογίζει τον αριθμό k που εμφανίζεται τουλάχιστον δύο φορές στον πίνακα A. Ο αλγόριθμός σας πρέπει να είναι *in-place*, δηλαδή πρέπει να χρησιμοποιεί μόνο $O(1)$ θέσεις μνήμης επιπλέον του πίνακα εισόδου $A[1 \dots n]$.

Άσκηση 5: Αθροίσματα Στοιχείων σε Συγκεκριμένο Διάστημα

(α) Δίνονται δύο *ταξινομημένοι* πίνακες $A[1 \dots n]$ και $B[1 \dots m]$, με n και m ακέραιους αριθμούς αντίστοιχα, και ένα ζευγάρι ακεραίων αριθμών L και M, με $L < M$. Ένα ζευγάρι θέσεων (i, j), με τη θέση i, $1 \leq i \leq n$, στον πίνακα A και τη θέση j, $1 \leq j \leq m$, στον πίνακα B, θεωρείται έγκυρο αν $L \leq B[j] - A[i] \leq M$. Να διατυπώσετε αλγόριθμο γραμμικού χρόνου που υπολογίζει το συνολικό πλήθος των έγκυρων ζευγαριών θέσεων (i, j) στους πίνακες A και B. Να αιτιολογήσετε συνοπτικά την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας.

(β) Δίνονται ένας *μη ταξινομημένος* πίνακας $A[1 \dots n]$ με n ακεραίους αριθμούς και ένα ζευγάρι ακεραίων αριθμών L και M, με $L < M$. Ένα ζευγάρι θέσεων (i, j) στον πίνακα A, με $1 \leq i < j \leq n$, θεωρείται έγκυρο αν $L \leq A[j] - A[i] \leq M$. Να διατυπώσετε αλγόριθμο χρόνου $O(n \log n)$ που υπολογίζει το συνολικό πλήθος των έγκυρων ζευγαριών θέσεων (i, j) στον πίνακα A. Να αιτιολογήσετε συνοπτικά την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας. *Υπόδειξη:* Μπορείτε να τροποποιήσετε κατάλληλα τον αλγόριθμο ταξινόμησης mergesort και να χρησιμοποιήσετε το αποτέλεσμα του (β) στη φάση της σύνθεσης των αποτελεσμάτων.

(γ) Δίνονται ένας *μη ταξινομημένος* πίνακας $A[1 \dots n]$ με n ακεραίους αριθμούς και ένα ζευγάρι ακεραίων αριθμών L και M, με $L < M$. Ένα διάστημα $A[i \dots j]$ στον πίνακα A, με $1 \leq i \leq j \leq n$, θεωρείται έγκυρο αν $L \leq \sum_{\ell=i}^{j} A[\ell] \leq M$. Να διατυπώσετε έναν αποδοτικό αλγόριθμο που υπολογίζει το συνολικό πλήθος των έγκυρων διαστημάτων $A[i \dots j]$ στον πίνακα A.