Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Αλγόριθμοι και Πολυπλοκότητα
Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης
1η Σειρά Γραπτών Ασκήσεων - Ημ/νία Παράδοσης 19/12/2013

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 $g_1 = O(g_2)$, $g_2 = O(g_3)$, κοκ. Σε αυτή τη διάταξη, να επισημάνετε τις συναρτήσεις που έχουν ίδια τάξη μεγέθους.

$5n$	$(\log n)^{\log n}$	$\log(n!)/n$	$n2^{2^{2^{100}}}$
$\log(\binom{n}{6})$	$\sum_{k=1}^{n} k^5$	$\log^4 n$	$\sqrt{n!}$
e^n	$n^2/\log^{10} n$	$(\log n)^{\log(16n)}$	$\log\left(\binom{2n}{n}\right)$
$n(2.5)^n$	$\binom{2n}{n}$	$\sum_{i=1}^{n} i2^i$	$\sum_{i=1}^{n} i2^{-i}$

(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. $T(n) = 5T(n/5) + n \log n$
2. $T(n) = 9T(n/10) + \log^3 n$
3. $T(n) = 2T(n/3) + n/\log^2 n$
4. $T(n) = T(n/6) + 3T(n/5) + n$
5. $T(n) = T(n/6) + T(n/2) + T(n/3) + n$
6. $T(n) = T(n-1) + 1/n$
7. $T(n) = 2T(\sqrt{n}) + \Theta(\log n)$
8. $T(n) = T(n-10) + \log n$.

Άσκηση 2: Ταξινόμηση

Έστω πίνακας θετικών ακεραίων $A[1 \ldots n]$ και έστω M το μέγιστο στοιχείο του A.

(α) Να δείξετε ότι ο A μπορεί να ταξινομηθεί σε χρόνο $O(n + M)$. Αν $M = O(n)$, ο χρόνος ταξινόμησης είναι γραμμικός. Γιατί δεν ισχύει το κάτω φράγμα του $\Omega(n \log n)$ σε αυτή την περίπτωση;

(β) Να δείξετε ένα κάτω φράγμα στο χρόνο εκτέλεσης χειρότερης περίπτωσης κάθε συγκριτικού αλγόριθμου για τον πίνακα A. Τι πρέπει να ισχύει για το M ώστε να υπάρχει συγκριτικός αλγόριθμος ταξινόμησης με χρόνο εκτέλεσης $O(n \log \log n)$; Για αυτές τις τιμές του M, να διατυπώσετε συγκριτικό αλγόριθμο ταξινόμησης με χρόνο εκτέλεσης $O(n \log \log n)$.

(γ) Έστω ότι $M = n^d - 1$, όπου d μια θετική ακέραια σταθερά. Να διατυπώσετε αλγόριθμο που ταξινομεί τον πίνακα A σε γραμμικό χρόνο. *Υπόδειξη:* Μπορείτε να θεωρήσετε ότι τα στοιχεία του A αναπαριστώνται στο n-αδικό σύστημα αρίθμησης και να επεκτείνετε τον αλγόριθμο του (α). Θα σας βοηθήσει ακόμη να δείτε πως λειτουργεί ο (μη συγκριτικός) αλγόριθμος ταξινόμησης Radixsort.

Άσκηση 3: Αναζήτηση

(α) Έστω k, n θετικοί φυσικοί, με $k < n$, και έστω πίνακας $A[0 \dots n]$ με $n+1$ φυσικούς μικρότερους ή ίσους του n. Υποθέτουμε ότι οι τιμές κάθε δύο διαδοχικών στοιχείων του A διαφέρουν το πολύ κατά k, δηλαδή ότι για κάθε j, $|A[j] - A[j+1]| \leq k$. (i) Να δείξετε ότι υπάρχει θέση j τέτοια ώστε $|A[j] - j| \leq (k+1)/2$. (ii) Να διατυπώσετε έναν αποδοτικό αλγόριθμο που υπολογίζει μια τέτοια θέση. Να προσδιορίσετε την υπολογιστική πολυπλοκότητα του αλγορίθμου σας και να αιτιολογήσετε την ορθότητά του.

(β) Έστω n, m θετικοί φυσικοί, με $m \leq n$, και έστω διδιάστατος πίνακας $A[1 \dots n, 1 \dots m]$ με nm φυσικούς. Γνωρίζουμε ότι τα στοιχεία του πίνακα A είναι ταξινομημένα σε γνήσια αύξουσα σειρά σε κάθε γραμμή του και σε κάθε στήλη του (δηλ. για κάθε i, j, ισχύει ότι $A[i,j] < A[i,j+1]$ και ότι $A[i,j] < A[i+1,j]$). Να διατυπώσετε έναν αποδοτικό αλγόριθμο που υπολογίζει όλες τις θέσεις του πίνακα A όπου εμφανίζεται μια δεδομένη τιμή k. Να προσδιορίσετε την υπολογιστική πολυπλοκότητα του αλγορίθμου σας (συναρτήσει των n και m) και να αιτιολογήσετε την ορθότητά του.

(γ) **Bonus ερώτημα:** Να αποδείξετε ένα ακριβές κάτω φράγμα στον χρόνο εκτέλεσης χειρότερης περίπτωσης κάθε συγκριτικού αλγόριθμου για το πρόβλημα (β). Είναι ο αλγόριθμος που διατυπώσατε βέλτιστος για όλα τα m, n, με $m \leq n$. Αν όχι, να διατυπώσετε έναν βέλτιστο αλγόριθμο.

Άσκηση 4: Επιλογή

(α) Έστω πολυσύνολο (multiset) S με n θετικούς ακέραιους που όλοι είναι μικρότεροι ή ίσοι δεδομένου ακεραίου M. Έχουμε πρόσβαση (μόνο) στην κατανομή F_S των στοιχείων της συλλογής. Συγκεκριμένα, έχουμε στη διάθεσή μας συνάρτηση $F_S(\ell)$ που για κάθε φυσικό ℓ, επιστρέφει το πλήθος των στοιχείων του S που δεν ξεπερνούν το ℓ, δηλ. $F_S(\ell) = |\{x \in S : x \leq \ell\}|$. Να διατυπώσετε αποδοτικό αλγόριθμο που δέχεται ως είσοδο φυσικό k, $1 \leq k \leq n$, και υπολογίζει (καλώντας την F_S) το k-οστό μικρότερο στοιχείο του S. Να αιτιολογήσετε την ορθότητα του αλγορίθμου σας και να προσδιορίσετε το πλήθος των απαιτούμενων κλήσεων στην F_S (στη χειρότερη περίπτωση). Προσπαθήστε το πλήθος των κλήσεων στην F_S να μην εξαρτάται από το n (μπορεί όμως να εξαρτάται από το M).

(β) Έστω πίνακας διαφορετικών θετικών ακεραίων $A[1 \dots n]$ και έστω M το μέγιστο στοιχείο του A. Θεωρούμε το πολυσύνολο S που αποτελείται από όλες τις μη αρνητικές διαφορές ζευγών στοιχείων του A. Δηλαδή, έχουμε ότι:

$$S = \{A[i] - A[j] : i \neq j \text{ και } A[i] > A[j]\}$$

Να διατυπώσετε αποδοτικό αλγόριθμο που υπολογίζει το k-οστό μικρότερο στοιχείο του S. Να προσδιορίσετε την υπολογιστική πολυπλοκότητα του αλγορίθμου σας (συναρτήσει των n και M) και να αιτιολογήσετε την ορθότητά του. *Υπόδειξη:* Προσπαθήστε να υλοποιήσετε αποδοτικά την F_S και να χρησιμοποιήσετε τον αλγόριθμο του (α).