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 5/12/2011

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.

$n3^n$	$n^{1.01}$	$5^{\log_2 n}$	$\sum_{k=1}^{n} k^5$
$2^{(\log_2 n)^4}$	$(\log n)^{\log n}$	$\frac{n}{\log\log n}$	$e^{n/\ln n}$
$\log(n^3)$	$\sqrt{n}(\log n)^{50}$	$n(\log n)^{10}$	$(\log n)^{\sqrt{n}}$
$n^{\log\log n}$	2^{2n}	$\sqrt{n!}$	$\log(n!)$

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

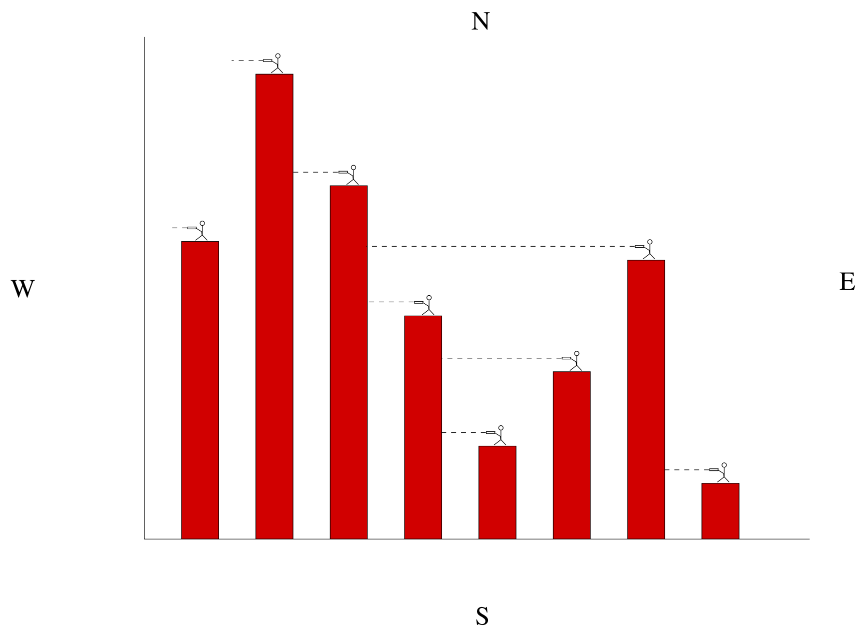
1. $T(n) = 5T(n/7) + n\log n$
2. $T(n) = 4T(n/5) + n/\log^2 n$
3. $T(n) = T(n/3) + 3T(n/7) + n$
4. $T(n) = 6T(n/6) + n$
5. $T(n) = T(n/3) + T(2n/3) + n$
6. $T(n) = 16T(n/4) + n^3\log^2 n$
7. $T(n) = T(\sqrt{n}) + \Theta(\log\log n)$
8. $T(n) = T(n-3) + \log n$.

Άσκηση 2: Ταξινόμηση σε Πίνακα με Πολλά ΄Ιδια Στοιχεία

΄Εστω πίνακας ακεραίων $A[1\ldots n]$ που χαρακτηρίζεται από πολλές πολλαπλές εμφανίσεις των στοιχείων του. Συγκεκριμένα, θεωρούμε ότι το πλήθος των διαφορετικών στοιχείων του A είναι μόλις πολυλογαριθμικό (δηλ. $O(\log^d n)$, για κάποια σταθερά $d \geq 1$). Να διατυπώσετε έναν συγκριτικό αλγόριθμο που ταξινομεί τον πίνακα A σε χρόνο $O(n\log\log n)$. Γιατί δεν ισχύει το κάτω φράγμα του $\Omega(n\log n)$ σε αυτή την περίπτωση;

Άσκηση 3: Δυαδική Αναζήτηση

(α) Μας δίνεται ένας πίνακας A με n ακέραιους ταξινομημένους σε αύξουσα σειρά, αλλά *δεν γνωρίζουμε την τιμή του n*. Ο τύπος δεδομένων του A έχει υλοποιηθεί ώστε να επιστρέφει το μήνυμα σφάλματος ∞ κάθε φορά που επιχειρείται η προσπέλαση στοιχείου $A[i]$ με $i > n$. Να διατυπώσετε αλγόριθμο με χρόνο εκτέλεσης $O(\log n)$, που δέχεται ως είσοδο έναν ακέραιο x, και βρίσκει μια θέση του πίνακα A που περιέχει τον x, αν υπάρχει τέτοια θέση.



Σχήμα 1. Ένα παράδειγμα για την Άσκηση 5 με 8 πολυκατοικίες. Το αποτέλεσμα είναι $B = [0, 0, 2, 3, 4, 4, 3, 7]$.

(β) Έστω δύο ταξινομημένοι πίνακες ακεραίων $A[1 \dots n]$ και $B[1 \dots n]$. Να διατυπώσετε αλγόριθμο με χρόνο εκτέλεσης $O(\log k)$ που υπολογίζει το k-οστό μικρότερο στοιχείο της ένωσης των πινάκων A και B. Για ευκολία, μπορείτε να υποθέσετε ότι όλα τα στοιχεία των A και B είναι διαφορετικά.

Άσκηση 4: Συλλογή Comics

Ένας φανατικός συλλέκτης comics θυμάται ότι από τη συλλογή του αγαπημένου του geek comic έχει χάσει ένα τεύχος, αλλά δεν θυμάται ποιο. Χρειάζεται να εντοπίσει το τεύχος αυτό, γιατί όλα τα τεύχη της σειράς πωλούνται στο Internet σε τιμή ευκαιρίας, σε μία δημοπρασία που λήγει άμεσα.

Ένας από τους λόγους που η συγκεκριμένη σειρά comics είναι η αγαπημένη του έχει να κάνει με τον ιδιόρρυθμο τρόπο αρίθμησης των τευχών: η αρίθμηση των τευχών είναι δυαδική, ξεκινά από το $0 \dots 0$ και ολοκληρώνεται στο $1 \dots 1$. Κάθε τεύχος έχει k σελίδες, το πλήθος των τευχών είναι $n = 2^k$, και τα bits του αριθμού κάθε τεύχους είναι γραμμένα από ένα σε κάθε σελίδα. Έτσι το μοναδικό που μπορεί να κάνει ο συλλέκτης για να εντοπίσει το τεύχος που λείπει είναι να εξετάζει κάθε φορά το i-οστό bit της αρίθμησης ενός τεύχους j.

Ο συλλέκτης χρειάζεται λοιπόν έναν αλγόριθμο που εντοπίζει το τεύχος που λείπει με όσο το δυνατόν λιγότερες ερωτήσεις της μορφής "ποιο είναι το i-οστό bit στην αρίθμηση του τεύχους j;". Μπορείτε να βοηθήσετε τον συλλέκτη διατυπώνοντας έναν αλγόριθμο που χρειάζεται το πολύ $2n$ τέτοιες ερωτήσεις;

Άσκηση 5: Πολυκατοικίες χωρίς Θέα

Κατά μήκος ενός πυκνοκατοικημένου δρόμου, που είναι χαραγμένος στη διεύθυνση ανατολή - δύση, υπάρχουν n πολυκατοικίες με διάφορα ύψη. Κατά μια περίεργη σύμπτωση, όλοι οι κάτοικου του δρόμου είναι αμετανόητα ρομαντικοί, και επιθυμούν να βλέπουν το ηλιοβασίλεμα από την ταράτσα της πολυκατοικίας τους. Καθώς όμως έχουν χτιστεί όλες αυτές οι πολυκατοικίες, κάτι τέτοιο δεν είναι εφικτό για τους περισσότερους. Θέλοντας λοιπόν να ρυθμίσουν την κατάσταση, αποφασίζουν την οργάνωση ενός είδους μονομαχίας (ως ρομαντικοί), ώστε να λυθεί το πρόβλημα. Εφοδιάζονται λοιπόν με όπλα, και αποφασίζουν το επόμενο πρωί να ανέβουν στις ταράτσες τους και να προσπαθήσουν να διαλύσουν την πολυκατοικία που έχει μεγαλύτερο ύψος από τη δική τους και εμποδίζει τη θέα της δύσης (ή την πρώτη που βλέπουν από αυτές, αν είναι πολλές). Τα όπλα τους ρίχνουν μόνο στην ευθεία παράλληλα με το έδαφος, και οι αποστάσεις μεταξύ των πολυκατοικιών είναι τέτοιες ώστε η βαρύτητα δεν παίζει κανένα ρόλο.

Ο Δήμαρχος πληροφορείται την κατάσταση, και αντιλαμβάνεται ότι πρέπει να ενεργήσει γρήγορα, αν θέλει να προλάβει τα χειρότερα. Για αυτό, προσπαθεί να υπολογίσει σε ποια πολυκατοικία θα σημαδέψουν εκείνο το πρωί οι κάτοικοι κάθε άλλης πολυκατοικίας. Αριθμεί τις πολυκατοικίες από τα δυτικά προς τα ανατολικά, και συμπληρώνει έναν πίνακα ακεραίων $A[1 \ldots n]$ με τα ύψη τους (το ύψος των ανθρώπων θεωρείται αμελητέο). Δεδομένου του πίνακα A, ο Δήμαρχος χρειάζεται να συμπληρώσει έναν πίνακα $B[1 \ldots n]$, που κάθε στοιχείο του $B[i]$ δίνει την πολυκατοικία που θα δεχθεί τα πυρά των κατοίκων της πολυκατοικίας i (αν όλες οι πολυκατοικίες δυτικά της i έχουν ύψος μικρότερο ή ίσο του $A[i]$, τότε $B[i] = 0$, βλ. ακόμη το παράδειγμα στο Σχήμα 1). Μπορείτε να βοηθήσετε το Δήμαρχο διατυπώνοντας έναν αλγόριθμο γραμμικού χρόνου για αυτό το πρόβλημα; Να επιμείνετε στην αναλυτική αιτιολόγηση του γραμμικού χρόνου εκτέλεσης του αλγορίθμου σας.