ΑΛΓΟΡΙΘΜΟΙ

Διδάσκουσα: Κ. Παπακωνσταντινοπούλου

1η Σειρά Ασκήσεων (Γραπτών)
Προθεσμία υποβολής: 15/6/2020

Τις λύσεις των ασκήσεων πρέπει να τις γράψετε σε ηλεκτρονική μορφή και να τις υποβάλετε στο eClass ως αρχείο .pdf με όνομα τον αριθμό μητρώου σας, δηλαδή 3xxxxxx.pdf.
Όσοι χρησιμοποιήσετε το σύστημα LaTeX, θα πρέπει να παραδώσετε και το .tex αρχείο, οπότε να υποβάλετε ένα συμπιεσμένο φάκελο .zip με όνομα τον αριθμό μητρώου σας, δηλαδή 3xxxxxx.zip, που θα περιέχει τα δύο αρχεία.

Άσκηση 1.1: Ασυμπτωτικός Συμβολισμός.

Σε κάθε μία από τις ακόλουθες περιπτώσεις, δείξτε αν είναι $f = O(g)$ ή $f = \Omega(g)$ ή και τα δύο, οπότε $f = \Theta(g)$.

	$f(n)$	$g(n)$
α.	$n^{1/2}$	$n^{3/4}$
β.	$100n + \log n$	$n + (\log n)^2$
γ.	$n2^n$	3^n
δ.	$\sum_{i=1}^{n} i^k$	$\sum_{k=1}^{n} k2^k$
ε.	$\sum_{k=1}^{n} \binom{n}{k}$	2^n
στ.	$n!$	2^n
ζ.	$n!$	n^n

Άσκηση 1.2: Υπολογισμός Πολυπλοκότητας.

Θεωρούμε το πρόβλημα ''Αθροισμα k-υποσυνόλου':
Είσοδος: Δίνεται σύνολο S θετικών ακεραίων και ακέραιοι k και M.
Ερώτηση: Υπάρχει υποσύνολο A του S με k στοιχεία, δηλαδή $|A| = k$, και άθροισμα στοιχείων $= M$;
καθώς και το πρόβλημα ''Αθροισμα υποσυνόλου':
Είσοδος: Δίνεται σύνολο S θετικών ακεραίων και ακέραιος M.
Ερώτηση: Υπάρχει υποσύνολο A του S με άθροισμα στοιχείων $= M$;

Για την επίλυση καθενός από τα προβλήματα αυτά θεωρούμε τον απλούστερο εξαντλητικό αλγόριθμο, δηλαδή τον αλγόριθμο που δημιουργεί συστηματικά πιθανές λύσεις και για κάθε μία ελέγχει αν είναι πράγματι λύση.

Να υπολογίσετε τις πολυπλοκότητες των δύο αλγορίθμων και να δώσετε τους κατάλληλους O-συμβολισμούς.

Άσκηση 1.3: Αναδρομικές Σχέσεις.

Εξετάζουμε τέσσερις διαφορετικούς αναδρομικούς αλγορίθμους για την επίλυση ενός υπολογιστικού προβλήματος Π.

α. Ο αλγόριθμος A_1 διασπά το αρχικό πρόβλημα μεγέθους n σε 4 υποπροβλήματα μεγέθους $n/4$, τα επιλύει και στη συνέχεια συνθέτει τις λύσεις τους σε χρόνο $7n$.

β. Ο αλγόριθμος A_2 διασπά το αρχικό πρόβλημα μεγέθους n σε 4 υποπροβλήματα μεγέθους $n/16$, τα επιλύει και στη συνέχεια συνθέτει τις λύσεις τους σε χρόνο $n^{5/4}$.

γ. Ο αλγόριθμος A_3 διασπά το αρχικό πρόβλημα μεγέθους n σε 8 υποπροβλήματα μεγέθους $n/4$, τα επιλύει και στη συνέχεια συνθέτει τις λύσεις τους σε χρόνο $6\sqrt{n}$.

δ. Ο αλγόριθμος A_4 διασπά το αρχικό πρόβλημα μεγέθους n σε 2 υποπροβλήματα μεγέθους $n-2$, τα επιλύει και στη συνέχεια συνθέτει τις λύσεις τους σε σταθερό χρόνο. Υποθέτουμε, χωρίς βλάβη της γενικότητας, ότι ο n είναι άρτιος και θεωρούμε ότι ο χρόνος επίλυσης για μηδενική είσοδο είναι ίσος με 1.

Ποιον αλγόριθμο από τους παραπάνω θα προτιμούσατε για την επίλυση του Π και γιατί;

Υπόδειξη: Για κάθε αλγόριθμο να γράψετε την αναδρομική εξίσωση που δίνει το χρόνο εκτέλεσής του και να υπολογίσετε την ασυμπτωτική πολυπλοκότητά του.

Άσκηση 1.4: Τεχνική Διαίρει-και-Βασίλευε

Μια αντιστροφή σε ένα πίνακα $A[1 \dots n]$ είναι ένα ζεύγος θέσεων (i, j) τέτοιο ώστε $i < j$ και $A[i] > A[j]$. Το πλήθος των αντιστροφών σε ένα πίνακα n στοιχείων κυμαίνεται μεταξύ 0 (αν ο πίνακας είναι ταξινομημένος σε αύξουσα σειρά) και $\binom{n}{2}$ (αν ο πίνακας είναι ανάποδα ταξινομημένος, δηλαδή ταξινομημένος σε φθίνουσα σειρά).

Να δώσετε έναν διαίρει-και-βασίλευε αλγόριθμο που δέχεται ως είσοδο ένα πίνακα n στοιχείων και υπολογίζει το πλήθος των αντιστροφών που περιέχει σε χρόνο $O(n \log n)$. Να αιτιολογήσετε την ορθότητα του αλγορίθμου σας και να υπολογίσετε την πολυπλοκότητά του.

Υπόδειξη: Μπορείτε να σχεδιάσετε τον ζητούμενο αλγόριθμο τροποποιώντας κατάλληλα τον αλγόριθμο Mergesort.

Άσκηση 1.5: Άπληστη τεχνική

Δίνονται n διαστήματα $[s_i, f_i]$ στον άξονα x, με $s_i, f_i \in \mathbf{R}$ και $s_i \leq f_i$, και ζητείται το *ελάχιστο πλήθος σημείων* της ευθείας που μπορούμε να επιλέξουμε έτσι ώστε να καλύψουμε όλα τα διαστήματα. Θεωρούμε ότι ένα διάστημα καλύπτεται αν περιέχει *τουλάχιστον ένα* σημείο από τα επιλεγμένα.

Να δώσετε άπληστο αλγόριθμο που υπολογίζει το ζητούμενο πλήθος σημείων, να αποδείξετε την ορθότητά του και να υπολογίσετε την πολυπλοκότητά του (δηλαδή να δώσετε την πολυπλοκότητά του αιτιολογώντας την απάντησή σας).

Υπόδειξη: Για την απόδειξη ορθότητας μπορείτε να χρησιμοποιήσετε το επιχείρημα της ανταλλαγής.

Άσκηση 1.6: Τεχνική Δυναμικού Προγραμματισμού

Θέλετε να βοηθήσετε τους υπαλλήλους μιας μεταφορικής εταιρείας να επιλέγουν τις σωστές κούτες για το πακετάρισμα των αντικειμένων, έτσι ώστε η μεταφορά να γίνεται χωρίς κίνδυνο φθοράς των αντικειμένων.

Υπάρχουν n αντικείμενα που πρέπει να μεταφερθούν, και κάθε αντικείμενο i έχει βάρος w_i. Η εταιρεία διαθέτει κούτες διαφόρων χωρητικοτήτων. Σύμφωνα με την εμπειρία των υπαλλήλων, η μεταφορά μιας κούτας που περιέχει αντικείμενα γίνεται με ασφάλεια, αν η κούτα περιέχει αντικείμενα συνολικού βάρους ακριβώς ίσου με τη χωρητικότητά της. Αν το βάρος είναι μεγαλύτερο, τότε η κούτα δε θα αντέξει και θα σκιστεί. Αν το βάρος είναι μικρότερο, αυτό σημαίνει ότι η κούτα δεν είναι γεμάτη, οπότε το βάρος δεν είναι ομοιόμορφα κατανεμημένο στον όγκο της και κατά τη μεταφορά είναι πιθανό να ανατραπεί. Οι υπάλληλοι θέλουν για κάθε διαθέσιμη κούτα να μπορούν να βρίσκουν το σύνολο αντικειμένων που πρέπει να τοποθετηθεί σε αυτή.

Να γράψετε αλγόριθμο που παίρνει ως είσοδο ένα μονοδιάστατο πίνακα ακεραίων W με τα βάρη των αντικειμένων, και έναν ακέραιο C που εκφράζει τη χωρητικότητα μιας κούτας, κι επιστρέφει TRUE *αν και μόνο αν* υπάρχει υποσύνολο των αντικειμένων με συνολικό βάρος *ακριβώς* C, με τη βοήθεια του *δυναμικού προγραμματισμού.*

Να δώσετε επίσης:

- την αναδρομική εξίσωση που χρησιμοποιήσατε για να σχεδιάσετε τον αλγόριθμο,
- την πολυπλοκότητα του αλγορίθμου σας, αιτιολογώντας την απάντησή σας.

Υπόδειξη: Το πρόβλημα αυτό είναι επί της ουσίας ένα πολύ γνωστό πρόβλημα, το έχουμε αναφέρει στο μάθημα και μοιάζει με το διακριτό πρόβλημα του σακιδίου (είναι μια απλούστερη εκδοχή του). Μπορείτε να βρείτε ποιο; Συνεπώς μπορείτε να σχεδιάσετε τον ζητούμενο αλγόριθμο τροποποιώντας κατάλληλα τον αλγόριθμο για το 0-1 Knapsack.

Άσκηση 1.7: Ερώτησης λύσης και ερώτηση απόφασης

Θεωρήστε το πρόβλημα εύρεσης ενός *κύκλου* Hamilton (HC):
Είσοδος: Γράφος $G = (V, E)$.
Έξοδος: Ένας *κύκλος* του G που διέρχεται από όλους τους κόμβους του V, ακριβώς μία φορά από τον καθένα, αν υπάρχει.

Στην εκδοχή απόφασης του προβλήματος αυτού (HC-D) ζητείται αν υπάρχει *κύκλος* Hamilton ή όχι:
Είσοδος: Γράφος $G = (V, E)$.
Έξοδος: 'Ναι' αν ο G έχει *κύκλο* Hamilton, διαφορετικά 'όχι'.

Δείξτε ότι αν έχετε αλγόριθμο πολυωνυμικού χρόνου για το HC-D, μπορείτε να τον χρησιμοποιήσετε για να σχεδιάσετε αλγόριθμο πολυωνυμικού χρόνου για το HC.

Βοήθεια: Έχουμε απαντήσει αυτό το ερώτημα στο μάθημα για παρόμοιο πρόβλημα.

Άσκηση 1.8: Πρόβλημα βελτιστοποίησης

Θεωρήστε το πρόβλημα του περιοδεύοντος πωλητή (TSP):
Είσοδος: Πλήρης γράφος $G = (V, E)$ με κόμβους που αναπαριστούν πόλεις και βάρη στις ακμές που δηλώνουν αποστάσεις μεταξύ των αντίστοιχων πόλεων, *και ένας προϋπολογισμός b*.
Έξοδος: Μια περιήγηση που διέρχεται από όλες τις πόλεις, ακριβώς μία φορά από την κάθε μία *και έχει μήκος το πολύ b*, αν υπάρχει τέτοια περιήγηση.

Στην εκδοχή βελτιστοποίησης του προβλήματος αυτού (TSP-OPT) ζητείται να βρεθεί η συντομότερη περιήγηση:
Είσοδος: Πλήρης γράφος $G = (V, E)$ με κόμβους που αναπαριστούν πόλεις και βάρη στις ακμές που δηλώνουν αποστάσεις μεταξύ των αντίστοιχων πόλεων.
Έξοδος: Η *συντομότερη* περιήγηση που διέρχεται από όλες τις πόλεις, ακριβώς μία φορά από την κάθε μία.

Δείξτε ότι αν μπορεί να λυθεί σε πολυωνυμικό χρόνο το πρόβλημα TSP, τότε μπορεί να λυθεί σε πολυωνυμικό χρόνο και το πρόβλημα TSP-OPT.

Βοήθεια: Έχουμε απαντήσει παρόμοιο ερώτημα στο μάθημα για το ίδιο πρόβλημα.