ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ — ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

ΣΧΟΛΗ ΕΠΙΣΤΗΜΩΝ & ΤΕΧΝΟΛΟΓΙΑΣ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ — SCHOOL OF INFORMATION SCIENCES & TECHNOLOGY — ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ — DEPARTMENT OF INFORMATICS

ΑΛΓΟΡΙΘΜΟΙ

Διδάσκουσα: Αλκμήνη Σγουρίτσα

2η Σειρά Ασκήσεων - Εαρινό Εξάμηνο 2023

Προθεσμία: **12 Ιουνίου 2023**

0.5 Μονάδες bonus στον τελικό βαθμό

Οι λύσεις των ασκήσεων πρέπει να γραφτούν σε υπολογιστή (πχ Word ή Latex), και να παραδοθούν στο **eclass** σε μορφή **pdf**. Η προθεσμία της παράδοσης είναι **12 Ιουνίου 2023**, και ώρα 11:59μμ.

Βαθμολογία: **10 μονάδες συνολικά**
(αντιστοιχούν σε **0.5 Μονάδες bonus** στον τελικό βαθμό)

Ασκήσεις

1. (**2 μον.**) Σας έχει ανατεθεί να διοργανώσετε μια δεξίωση στον χώρο εργασίας σας και θα πρέπει να επιλέξετε ένα υποσύνολο από τους n εργαζόμενους της εταιρείας που θα καλέσετε. Για να είναι επιτυχημένη η δεξίωση έχετε σκεφθεί ότι πρέπει να ικανοποιούνται οι παρακάτω συνθήκες:

   (α΄) Ο κάθε καλεσμένος θα πρέπει να γνωρίζει **τουλάχιστον 4** άλλους προσκεκλημένους, ώστε να αισθάνεται άνετα στην δεξίωση.

   (β΄) Για κάθε προσκεκλημένο θα πρέπει να υπάρχουν **τουλάχιστον 4** άλλοι προσκεκλημένοι που δεν γνωρίζει, ώστε να δημιουργηθούν νέες γνωριμίες κατά τη διάρκεια της δεξίωσης.

   Ο στόχος σας είναι να προσκαλέσετε όσο πιο πολλά άτομα είναι δυνατόν, ώστε να ικανοποιούνται και οι δυο παραπάνω συνθήκες για κάθε καλεσμένο.

   Να δοθεί περιγραφικά πολυωνυμικός αλγόριθμος για το παραπάνω πρόβλημα. Η είσοδος του προβλήματος είναι ένας $n \times n$ πίνακας A (για n άτομα) με τιμές 0 και 1, όπου $A(i,j) = 1$ αν και μόνο αν ο i και ο j γνωρίζονται μεταξύ τους. Να επεξηγηθεί λεπτομερώς η ορθότητα και η πολυπλοκότητα ως προς n του αλγορίθμου σας.

2. (**2 μον.**) Θεωρήστε το πρόβλημα coin changing που κάναμε στο μάθημα, με την παραλλαγή ότι το κάθε νόμισμα μπορεί να χρησιμοποιηθεί το πολύ μία φορά και δεν κάνουμε καμία υπόθεση για το νόμισμα με τη μικρότερη αξία. Δηλαδή το πρόβλημα διατυπώνεται ως εξής:

   Είσοδος: n νομίσματα με αξίες $v_1 \geq v_2 \geq \ldots \geq v_n \geq 1$ και ένα ποσό E

   Έξοδος: Ο αλγόριθμος πρέπει να επιστρέφει "Αδύνατο", αν δεν υπάρχει σύνολο από νομίσματα, με το καθένα να εμφανίζεται **το πολύ μία φορά**, και με άθροισμα αξιών ίσο με E. Σε αντίθετη περίπτωση πρέπει να επιστρέφει το ελάχιστο πλήθος των νομισμάτων με συνολική αξία E, όπου το κάθε νόμισμα μπορεί να χρησιμοποιηθεί το **πολύ μία φορά**, και ποια είναι αυτά τα νομίσματα.

   Για παράδειγμα, αν σας δίνονται τα νομίσματα 3, 5, 8,10, 20 και το ποσό $E = 18$, τότε υπάρχουν δύο τρόποι να δημιουργήσουμε το άθροισμα 18, είτε με τα νομίσματα 3,5,10, είτε με τα νομίσματα 8,10. Ο αλγόριθμός σας θα πρέπει να επιστρέφει τη 2η λύση ως τη λύση με το ελάχιστο πλήθος νομισμάτων. Αν για τα ίδια νομίσματα το ποσό ήταν $E = 17$, ο αλγόριθμος θα πρέπει να επιστρέφει "Αδύνατο", αφού δεν υπάρχει τρόπος να δημιουργήσουμε το άθροισμα 17 χωρίς να χρησιμοποιήσουμε κάποια νομίσματα πάνω από μία φορά.

   Να δοθεί αλγόριθμος σε ψευδογλώσσα για το παραπάνω πρόβλημα (ο αλγοριθμός σας μπορεί να μην είναι πολυωνυμικός). Να επεξηγηθεί λεπτομερώς η ορθότητα (αρκεί να εξηγήσετε πώς προέκυψε η αναδρομική σχέση όπως κάναμε στα παραδείγματα στο μάθημα) και η πολυπλοκότητα του αλγορίθμου.

   Υπόδειξη: Η λογική σχετίζεται με το 0-1 Knapsack πρόβλημα που κάναμε στο μάθημα.

3. (**2 μον.**) Προγραμματίζετε ένα μεγάλο ταξίδι με αυτοκίνητο. Ξεκινάτε από ένα σημείο που βρίσκεται στη χιλιομετρική θέση $a_0 = 0$ και θέλετε να καταλήξετε σε ένα σημείο που βρίσκεται στη χιλιομετρική θέση a_n. Στη διαδρομή που θα ακολουθήσετε υπάρχουν n σημεία για στάση (συμπεριλαμβανομένου του σημείου προορισμού) στις χιλιομετρικές θέσεις, $0 < a_1 < a_2 < \ldots < a_n$, όπου a_i είναι η χιλιομετρική απόσταση από το σημείο εκκίνησης. Θεωρήστε τη διαδρομή σαν μία ευθεία, δηλαδή η απόσταση μεταξύ οποιονδήποτε δύο στάσεων i και j είναι $|a_i - a_j|$. Οι μοναδικές θέσεις όπου επιτρέπεται να κάνετε στάση είναι αυτές οι n τοποθεσίες. Μπορείτε όμως να επιλέξετε σε ποιες από αυτές τις στάσεις θα σταματήσετε. Πρέπει όμως οπωσδήποτε να σταματήσετε στο σημείο που βρίσκεται σε απόσταση a_n που είναι και ο τελικός σας προορισμός.

   Ιδανικά θα θέλατε να ταξιδεύετε συνεχόμενα για 40 χιλιόμετρα από τη μία στάση που θα κάνετε ως την επόμενη στάση που θα κάνετε, αλλά αυτό μπορεί να μην είναι εφικτό. Αν ταξιδέψετε x συνεχόμενα χιλιόμετρα από τη μία στάση που θα κάνετε ως την επόμενη στάση που θα κάνετε, η ποινή για τη συγκεκριμένη απόσταση είναι $(40-x)^2$. Θέλετε να προγραμματίσετε το ταξίδι σας ώστε να ελαχιστοποιήσετε τη συνολική ποινή, δηλαδή το άθροισμα των ποινών για κάθε διαδρομή χωρίς στάση, μέχρι να φτάσετε στον προορισμό σας. Υποθέστε ότι έχετε αρκετή βενζίνη για να κάνετε όλη τη διαδρομή από το σημείο εκκίνησης μέχρι τον προορισμό σας (δηλαδή a_n χιλιόμετρα) χωρίς στάση.

   Να σχεδιάσετε έναν πολυωνυμικό αλγόριθμο ο οποίος να προσδιορίζει τη βέλτιστη ακολουθία των στάσεων στις οποίες πρέπει να σταματήσετε. Να δώσετε τον αλγόριθμο περιγραφικά και σε ψευδογλώσσα. Να επεξηγηθεί λεπτομερώς η ορθότητα και η πολυπλοκότητα του αλγορίθμου.

4. Δίνεται μία σκακιέρα μεγέθους $n \times n$ όπου κάθε κελί έχει μία αξία (π.χ. μπορείτε να υποθέσετε ότι σε κάθε κελί έχει πιθανόν τοποθετηθεί κάποιο χρηματικό ποσό). Οι αξίες αυτές δίνονται σε ένα δισδιάστατο $n \times n$ πίνακα V με μη αρνητικές τιμές, δηλαδή η αξία του κελιού που βρίσκεται στη σειρά i και στη στήλη j της σκακιέρας είναι $V[i, j] \geq 0$. Σας ζητήθηκε να διαλέξετε ακριβώς ένα κελί από κάθε σειρά της σκακιέρας (άρα συνολικά n κελιά) με τον παρακάτω περιορισμό:

   Περιορισμός: Για κάθε δύο διαδοχικές σειρές δεν μπορείτε να διαλέξετε κελιά που ανήκουν στην ίδια στήλη. Δηλαδή, δεν επιτρέπεται να διαλέξετε τα κελιά $(1, 2)$, $(2, 2)$ και $(3, 1)$, γιατί στις διαδοχικές σειρές 1 και 2 τα επιλεγμένα κελιά ανήκουν και τα δύο στη 2η στήλη. Όμως επιτρέπεται να διαλέξετε τα κελιά $(1, 1)$, $(2, 2)$ και $(3, 1)$, γιατί τα κελιά που ανήκουν στην ίδια στήλη (δηλαδή τα $(1, 1)$ και $(3, 1)$) δεν ανήκουν σε διαδοχικές σειρές.

   Επιλέγετε τα κελιά με σκοπό τη μεγιστοποίηση της συνολικής τους αξίας. Σκέφτεστε δύο τρόπους προσέγγισης:

   - Τη χρήση άπληστου (greedy) αλγορίθμου.
   - Τη χρήση δυναμικού προγραμματισμού.

   Για την πρώτη προσέγγιση, έχετε σκεφτεί τον εξής greedy αλγόριθμο: Κάθε φορά επιλέγετε το κελί με τη μέγιστη αξία (από όλη τη σκακιέρα) και σέβεται τους περιορισμούς (δηλαδή δεν έχει επιλεγεί άλλο κελί στην ίδια σειρά και δεν έχει επιλεγεί κάποιο από τα κελιά που βρίσκονται στην ίδια στήλη και στην προηγούμενη (αν υπάρχει) ή στην επόμενη (αν υπάρχει) σειρά).

   (α΄) **(1 μον.)** Σχεδιάστε πολυωνυμικό αλγόριθμο με την τεχνική του δυναμικού προγραμματισμού. Ποια είναι η αναδρομική σχέση; Να δοθεί ο αλγόριθμος περιγραφικά και σε ψευδογλώσσα.

   (β΄) **(1 μον.)** Υπολογίστε την πολυπλοκότητα για καθέναν από τους δύο αλγορίθμους.

   (γ΄) **(2 μον.)** Για καθέναν από τους δύο αλγορίθμους, αναφέρετε αν δίνει τη βέλτιστη λύση ή όχι. Στην περίπτωση που δίνει τη βέλτιστη λύση, αποδείξτε την ορθότητα. Σε περίπτωση που δε δίνει τη βέλτιστη λύση, υπολογίστε πόσο μακριά είναι από τη βέλτιστη λύση στη χειρότερη περίπτωση, δηλαδή αν για το στιγμιότυπο I ο αλγόριθμος δίνει συνολική αξία $A(I)$ και η βέλτιστη λύση δίνει συνολική αξία $B(I)$, υπολογίστε το $\alpha = \max_I \frac{B(I)}{A(I)}$, δηλαδή τον χειρότερο (μέγιστο) λόγο $\frac{B(I)}{A(I)}$ για όλα τα στιγμιότυπα I. Δικαιολογήστε και σε αυτή την περίπτωση την απάντησή σας και δώστε ένα από τα χειρότερα στιγμιότυπα, δηλαδή ένα στιγμιότυπο που ο αλγόριθμος επιστρέφει συνολική αξία (που τείνει στο) $1/\alpha$ της μέγιστης δυνατής.

   Επεξήγηση: Σαν στιγμιότυπο θεωρείται μία είσοδος του προβλήματος, δηλαδή ένας πίνακας V. Διαφορετικοί πίνακες V δίνουν διαφορετικά στιγμιότυπα.