Αλγόριθμοι και Πολυπλοκότητα

3η Σειρά Γραπτών Ασκήσεων

Ονοματεπώνυμο: Λιαροκάπης Αλέξανδρος
Αριθμός Μητρώου: 03114860

Άσκηση 1

(1.) Παρατηρούμε πως ο μέγιστος αριθμός ζευγών είναι min(Πομποί, Δέκτες). Έστι μπορούμε με δύο μετρητές και με μία γραμμική διάσχηση να βρούμε τον μέγιστο αριθμό ζευγών πετυχαίνοντας γραμμική πολυπλοκότητα.

(2.) Έστω $C(i, j)$ το κόστος της βέλτιστης λύσης μέχρι και την i κεραία με j πομπούς που δεν έχουν αντιστοιχηθεί με δέκτες. Τότε έχουμε την παρακάτω αναδρομική σχέση: $C(i, j) = \min C(i-1, j+1) + R_i, C(i-1, j-1) + T_i$. Η λύση θα είναι το $C(n, 0)$. Μπορούμε να υπολογίσουμε όλες τις $C(i, j)$ σε $O(n^2)$ χρόνο με δυναμικό προγραμματισμό.

Άσκηση 2

Μπορούμε να εφαρμόσουμε Dijstra με την κατάλληλη επιλογή γειτόνων (κόμβοι απόστασης 5). Τον υπολογισμό γειτόνων μπορούμε να τον κάνουμε κρατώντας έναν πίνακα γειτνίασης. Με 5 επαναλήψεις μπορούμε να κατασκευάσουμε γειτνίασης απόστασης 5. Κάθε επανάληψη θα έχει πολυπλοκότητα $O(n^3)$ η οποία και υπερισχύει της πολυπλοκοτητας του Dijstra. Έτσι ο συνολικός αλγόριθμος θα έχει πολυπλοκότητα $O(n^3)$.

Άσκηση 3

Μπορούμε να λύσουμε το πρόβλημα με τον παρακάτω ψευδοκώδικα:

```
rents_ending_at_[n] : each date is assigned all rents ending at that date O(m) construction
dp[n] : the best gain for each date

dp[0] = 0
for date from 1 to n:
   dp[date] = dp[date-1]
   for every rent in rents_ending_at(date):
        dp[date] = max(dp[date], dp[start_date(rent)] + gain(rent))
```

Ο εξωτερικός βρόγχος θα τρέξει n φορές και ο εσωτερικός συνολικά m φορές. Μαζί με την αρχικοποίηση του πίνακα τελικών ημερομηνιών ο αλγόριθμος έχει πολυπλοκότητα $O(n + m)$

Άσκηση 4

Αρχικά υπολογίζουμε πίνακες γειτνίασης μήκους 1,2 και 3. Αυτή η διαδικασία έχει πολυπλοκότητα $O(n^3)$. Έτσι μπορούμε να βρούμε όλους τους γείτονες απόστασης το πολύ 3 και αφού φιλτράρουμε κατάλληλα όσους έχουν σταθμό επισκευής, μπορούμε να εκτελέσουμε Dijstra για να λύσουμε το πρόβλημα. Η συνολική πολυπλοκότητα είναι $O(n^3)$.