

Αλγόριθμοι και προχωρημένες δομές δεδομένων
ΣΕΤ ΑΣΚΗΣΕΩΝ #4

Ημερομηνία παράδοσης 8 Φεβρουαρίου 2019

Χρυσόστομος Στύλιος και Χρήστος Γκόγκος

11/01/2019

Άσκηση 1

Δημιουργήστε έναν λογαριασμό στην ιστοσελίδα exercism (<https://exercism.io>). Εγγραφείτε στο track για τη γλώσσα C++. Επιλύστε τα ακόλουθα προβλήματα:

- ETL (easy)
- Queen Attack (medium)
- Say (hard)

Άσκηση 2

Το πρόβλημα 0-1 knapsack (0-1 σακιδίου) αφορά ένα σύνολο από αντικείμενα για τα οποία γνωρίζουμε το βάρος και την αξία κάθε αντικειμένου. Ζητείται η επιλογή ενός υποσυνόλου των αντικειμένων έτσι ώστε το συνολικό βάρος από τα επιλεγμένα αντικείμενα να μην ξεπερνά μια συγκεκριμένη τιμή βάρους και ταυτόχρονα να έχει επιτευχθεί η μεγαλύτερη δυνατή αξία. Το πρόθεμα 0-1 στο όνομα του προβλήματος υποδηλώνει ότι κάθε αντικείμενο μπορεί είτε να επιλεγεί είτε να μην επιλεγεί στο σύνολό του και όχι τμηματικά.

1. Δημιουργήστε στιγμιότυπα προβλημάτων 0-1 knapsack χρησιμοποιώντας τον generator του άρθρου [1] και τον κώδικα που βρίσκεται στη διεύθυνση (<http://hjemmesider.diku.dk/pisinger/generator.c>). Δημιουργήστε από 5 στιγμιότυπα για κάθε συνδυασμό των ακόλουθων παραμέτρων: $n=\{10,50,100,500\}$, $r=\{50,100,500,1000\}$ και $type=\{1,2,3,4\}$, δηλαδή σύνολο $5 \times 4 \times 4 \times 4 = 320$ στιγμιότυπα.
2. Χρησιμοποιώντας δυναμικό προγραμματισμό γράψτε σε C++ έναν επιλυτή για το πρόβλημα. Καταγράψτε το αποτέλεσμα και το χρόνο που χρειάστηκε για να επιλυθεί το κάθε πρόβλημα με τον επιλυτή σας.
3. Χρησιμοποιώντας το λογισμικό Google OR-Tools (<https://developers.google.com/optimization/>) και τον εξειδικευμένο επιλυτή που διαθέτει για προβλήματα knapsack επιλύστε τα στιγμιότυπα προβλημάτων που δημιουργήσατε. Καταγράψτε το αποτέλεσμα και το χρόνο που χρειάστηκε για να επιλυθεί το κάθε πρόβλημα.

Ο κώδικας να ανέβει στο GitHub.

Αναφορές

[1] Pisinger, David. Core problems in knapsack algorithms. Operations Research 47.4, pp. 570-575, 1999.