
Το πρόβλημα της μέγιστης υποακολουθίας, θεωρητική και εμπειρική μελέτη.

Όνομα φοιτητή
8 Δεκεμβρίου 2021

Περίληψη

Η πολυπλοκότητα ενός αλγορίθμου προσδιορίζει τον χρόνο που θα χρειαστεί για μία συγκεκριμένη διεργασία. Σε αυτήν την τεχνική αναφορά εξετάζεται το πρόβλημα της μέγιστης υποακολουθίας και γίνεται σύγκριση μεταξύ τριών διαφορετικών αλγορίθμων για την επίλυση του.

1 Εισαγωγή

Το πρόβλημα της μέγιστης υποακολουθίας (maximum subarray) είναι ένα πρόβλημα στο οποίο δίνεται μια ακολουθία τιμών (αρνητικές, θετικές ή μηδέν) και ζητείται να βρεθεί η συνεχόμενη σειρά τιμών της ακολουθίας που δίνει το μεγαλύτερο δυνατό άθροισμα.

Ο απλοϊκός τρόπος επίλυσης είναι πρακτικά ένας αλγόριθμος ωμής δύναμης με πολυπλοκότητα $\mathcal{O}(n^3)$. Ο βελτιωμένος αλγόριθμος δεν υπολογίζει ξανά όλα τα αθροίσματα αφού τα έχει υπολογίσει εκ' των προτέρων και η πολυπλοκότητα του είναι $\mathcal{O}(n^2)$. Τέλος ο αλγόριθμος του kadane είναι ο αποδοτικότερος τρόπος για την επίλυση με πολυπλοκότητα $\mathcal{O}(n)$.

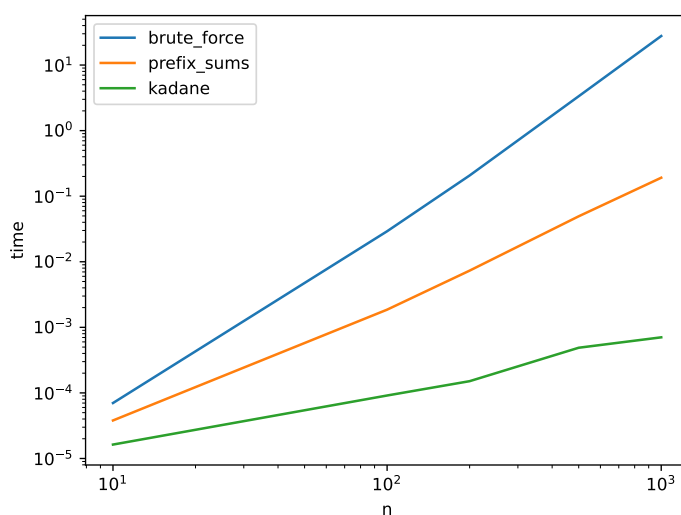
2 Αποτελέσματα

Για την συγγραφή των πειραμάτων χρησιμοποιήθηκε η python 3.9.7 και το Visual Studio Code. Τα χαρακτηριστικά του υπολογιστή είναι i7 3632QM (2.2 GHz), 8 GB DDR3 (1600 MHz). Για την οπτικοποίηση των αποτελεσμάτων χρησιμοποιήθηκε η βιβλιοθήκη matplotlib. Στον πίνακα 2.1 υπάρχουν οι χρόνοι για κάθε αλγόριθμο. Στην γραφική παράσταση 2.1 φαίνεται η σύγκριση των αλγορίθμων σε λογαριθμική κλίμακα.

Αλγόριθμος	Χρόνος (s)				
	n = 10	n = 100	n = 200	n = 500	n = 1000
Ωμής δύναμης	7.10E-05	0.028544	0.199864	3.322954	28.34849
Βελτιωμένος	3.48E-05	0.001919	0.007412	0.047613	0.184395
Kadane	2.45E-05	7.51E-05	0.000134	0.000316	0.000691

Πίνακας 2.1: Πίνακας χρονομετρήσεων

Σχήμα 2.1: Αποτελέσματα σε λογαριθμική κλίμακα.



3 Συμπεράσματα

Η σύγκριση του χρόνου των αλγορίθμων μας δείχνει ξεκάθαρα τις επιπτώσεις που έχει η χρήση ενός μη αποδοτικού αλγορίθμου για κάποιο πρόβλημα στο οποίο υπάρχουν πιο αποδοτικές λύσεις.